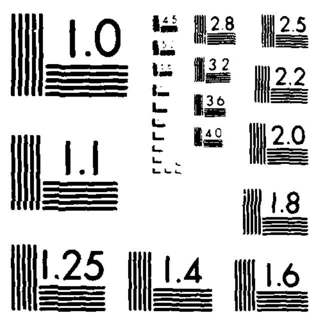
AD-A144 480 COMPUTER PROGRAM TO SIMULATE THE LATERAL-DIRECTIONAL RESPONSE OF A HIGH P..(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA S F GRAVES MAR 84 1/1

UNCLASSIFIED F/G 1/3 NL

END
DATE
FILMED
9-84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(2)

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

COMPUTER PROGRAM TO SIMULATE THE
LATERAL-DIRECTIONAL RESPONSE OF A HIGH
PERFORMANCE AIRCRAFT DIGITAL ELECTRONIC
FLIGHT CONTROL SYSTEM

by

Scott Friedrich Graves

March 1984

Thesis Advisor: Marle D. Hewett

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A144 410	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Computer Program to Simulate the Lateral-Directional Response of a High Performance Aircraft Digital Electronic Flight Control System		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1984
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Scott Friedrich Graves		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE March 1984
		13. NUMBER OF PAGES 76
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital Flight Control System Model
Lateral Control Laws
Directional Control Laws
Continuous System Modeling Program
Gain Schedule

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The IBM Company's Continuous System Modeling Program was used to simulate the lateral and directional flight control systems of the F/A-18 aircraft. The model is designed for use in studies of high angle-of-attack maneuvering flight and is restricted to the Auto Flaps Up mode of operation. The model accepts simulated pilot stick and rudder inputs, air data information, and rate gyro, angle-of-attack, and acceleration feedback

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

signals. Outputs are differential stabilator, differential leading-edge and trailing-edge flap, aileron, and rudder deflections. Typical input values are used to validate the model, generating output control surface deflections which correspond to those expected for the F/A-18 aircraft.

Approved for public release; distribution unlimited.

Computer Program to Simulate the Lateral-Directional Response of a High Performance Aircraft Digital Electronic Flight Control System

by

Scott Friedrich Graves
Lieutenant, United States Navy
B.S., University of Washington, 1975

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

Author: S. Graves

Approved by: Mark D. Hewett
Thesis Advisor

Donald M. Layton
Chairman, Department of Aeronautics

J. N. Dyer
Dean of Science and Engineering

ABSTRACT

The IBM Company's Continuous System Modeling Program was used to simulate the lateral and directional flight control systems of the F/A-18 aircraft. The model is designed for use in studies of high angle-of-attack maneuvering flight and is restricted to the Auto Flaps Up mode of operation. The model accepts simulated pilot stick and rudder inputs, air data information, and rate gyro, angle-of-attack, and acceleration feedback signals. Outputs are differential stabilator, differential leading-edge and trailing edge flap, aileron, and rudder deflections. Typical input values are used to validate the model, generating output control surface deflections which correspond to those expected for the F/A-18 aircraft.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

In this thesis, work completed at the Department of Aeronautics of the Naval Postgraduate School as part of that Department's research into the development of advanced control concepts using digital electronic flight control in U. S. Navy tactical combat aircraft is described. In concurrent work, Carter [Ref. 1], discusses in part the motivation for and the scope of this research program. Briefly reviewing Reference 1, it is desired to expand the application of digital flight control technology to the following areas:

1. Investigation of active control prevention of departure from controlled flight.
2. Comparative testing of new control law algorithms.
3. Evaluation of modern control techniques such as optimal control, observers, and model following control.
4. Simulation of degraded flight conditions for combat survivability studies.

As discussed in Reference 1, computer simulation of a digital fly-by-wire aircraft was deemed the best initial approach in studying the topics listed above. The Navy/McDonnell Douglas F/A-18 is such an aircraft, and in fact is the first U. S. aircraft in production to use a digital fly-by-wire control system. In Reference 1, the computer

modeling of the longitudinal axis control system of the F/A-18 is described. In this thesis, the model is extended to the lateral-directional axes to complete the control system model. Future incorporation of an F/A-18 aerodynamics simulation by Raithel [Ref. 2] and non-linear equations of motion will result in a complete "flying" computer model of the F/A-18.

The computer model was developed on the Naval Postgraduate School's IBM Model 3033 general purpose mainframe computer. The code is written for the IBM Company's Continuous System Modeling Program (CSMP) [Ref. 3], which is a Fortran application program designed to simulate dynamic systems. The text, "A Guide To Using CSMP" by Speckhart and Green [Ref. 4], provides most of the documentation necessary to use CSMP. CSMP has several advantages over a conventional Fortran program for this simulation. First, CSMP includes 34 built-in functions which serve to model most of the block diagrams encountered in control systems engineering. These functions are analogous to Fortran functions. Second, CSMP can accommodate user-defined functions, called macros, and can also call standard Fortran subroutines and functions. Third, output formatting, either printed or graphical, is handled by CSMP using just a few statements. Finally, the time base and numerical integration routines are provided by CSMP. One might consider the primary disadvantage of CSMP to be lack of user control over the built-in functions

and numerical methods. For this simulation CSMP proved to be a very effective tool.

In Chapter II, an overview of the F/A-18 flight control system is provided and the assumptions made and limitations of the modeling process are discussed. In Chapter III, the methodology and nomenclature of the computer program are outlined. In Chapter IV, the tests and results used to validate the model are presented. Conclusions and recommendations are in Chapter V. Appendix A contains block diagrams of the F/A-18 flight control system as modeled, including the previous longitudinal model developed by Carter [1]. Changes in nomenclature and arrangement to Carter's model have been made in incorporating it into the current model; these are essentially cosmetic. Appendix B is the computer program listing in CSMP of the F/A-18 flight control system as modeled, again, readers of Carter's program will note some changes in its form here to suit the overall program.

II. F/A-18 FLIGHT CONTROL SYSTEM

The F/A-18 flight control system is described in detail in the McDonnell Aircraft Company's F/A-18A Flight Control System Design Report [Ref. 5]. In this chapter, basics of the system are discussed and portions used in the model are indicated.

The flight control system is a digital fly-by-wire electronic control augmentation system which uses a four-channel parallel network of computers, electronic circuit elements, and associated wiring. The system is entirely electronic from pilot controls/feedback sensors to the control surface actuators. The actuators are redundant electrohydraulic servo mechanisms with the exception of the leading-edge flap system which is a hydraulically-powered rotary mechanical system. Backup mechanical control of the stabilators is available, and the stabilator, aileron and rudder surfaces have a backup analog Direct Electric Link in the event of a total digital computer failure. Control surfaces are stabilators, ailerons, dual rudders, leading-edge and trailing-edge flaps. The stabilators, leading-edge and trailing-edge flaps are capable of differential movement.

Pilot inputs are through a conventional control stick and rudder pedal arrangement. Closed-loop stability augmentation is provided by feedback of pitch, roll, and

yaw rates, normal and lateral acceleration, and angle-of-attack data. These feedback signals are gain scheduled by air data and angle-of-attack information to tailor the feedback signals to the current flight condition. Cross-axis control signal interconnects are provided which improve control and feedback coordination and reduce inertia coupling. Reference 5 should be consulted for a more in-depth discussion of the feedback and gain schedule design theory.

Initially, the computer model is to be used for studies involving only the "up and away" flight mode, meaning that the simulated aircraft is in a trimmed, stable condition in normal flight prior to the initiation of a maneuver. This allows the model to be reduced from the full control system. The assumptions which were made in reducing the model are discussed in Reference 1, however, several will be repeated here:

1. The aircraft is in the Auto Flaps Up configuration. The control law gain schedules used are designed for cruise and combat maneuvering flight. The leading and trailing-edge flaps are automatically positioned.
2. Inner loop control is being used. This means that the pilot is the source of control inputs through the stick and rudders. Outer loop (autopilot) control is not modeled.
3. The Control Augmentation System is in use. This is the optimum situation indicating that the digital computers,

feedback sensors, and air data, acceleration, and angle-of-attack measurements are all operating normally.

4. Trim, external stores, speedbrakes, and anti-spin features are not modeled. Active Oscillation Control is also left out, as it is a function of external store loading.

The model thus used is acceptable for simulating the full range of combat maneuvering flight of which the F/A-18 is capable. Take-off and approach/landing phases are not simulated.

In the McDonnell Aircraft Company's report, F/A-18 Flight Control Electronic Set Control Laws [Ref. 6], the software design and nomenclature of the F/A-18 digital control system is described. Those portions which are included in the model are briefly reviewed below, including for completeness the longitudinal portion which was discussed in Reference 1. Readers desiring in-depth information of the F/A-18 flight dynamics and control system theory of operation should refer to References 5 and 6.

LONGITUDINAL AXIS

The control system consists of the following paths:

- o Stick pitch input
- o Pitch rate feedback
- o Normal accelerometer feedback
- o Angle-of-attack feedback

- Inertial decoupling feedback (roll rate * yaw rate)
- Forward loop integrator

These paths are summed to form the stabilator command. Except for inertial decoupling feedback, all of the paths are used as inputs to the forward loop integrator path, which reduces the steady-state difference between maneuver command and aircraft response to zero. The signal paths are gain scheduled by air data, angle-of-attack, and acceleration functions which are discussed in Chapter III. A digital notch filter is located after the feedback summing junction to attenuate structural vibrations. In the Auto Flap Up mode, gain-scheduled angle-of-attack information is used to provide maneuvering flap commands.

LATERAL AXIS

The control system consists of the following paths:

- Stick roll input
- Roll rate feedback
- Rudder pedal interconnect

The lateral command is formed by the sum of the signal paths, which are gain-scheduled similarly to the longitudinal axis paths. The stick and roll rate paths incorporate digital notch filters to attenuate structural vibration inputs. Differential stabilator, and leading-edge and trailing edge flap commands are separately gain-scheduled and directed to their respective paths in the longitudinal axis control system.

DIRECTIONAL AXIS

The control system consists of the following paths:

- o Rudder pedal yaw input
- o Yaw rate feedback
- o Lateral acceleration feedback
- o Rolling surface interconnect
- o Inertial decoupling feedback (pitch rate * roll rate)

The rudder command is the sum of structural-filtered, gain scheduled feedback signals, and gain-scheduled rudder pedal and rolling surface interconnect signals.

Actuator data from Reference 5 was not expressed in terms of damping and natural frequency, therefore, data that was available on an actuator very much like the F/A-18 stabilator actuator was used as a guide. It was decided to use a second order model for all of the control surface actuators, and to use the same damping ratio and natural frequency, 0.7 and 40Hz, respectively, until the data specified in that manner was obtained for the other actuators.

Digital filters other than the structural filters already mentioned include lag, lead-lag, and integrators. The constants for all of the filters are listed in Chapter 16 of Reference 5. Aliasing filters were treated as analog filters (s-domain) as they always occur prior to A/D converters. Digital filters used the z-domain representation.

In Chapter III, the modeling methodology by which the F/A-18 digital control system as described briefly above was coded for computer study using CSMP is discussed.

III. PROGRAM METHODOLOGY

The primary source of information used in making the computer model was the McDonnell Aircraft Company F/A-18 Flight Control System Design Report [Ref. 5]. Figures 16.1, 16.2, and 16.3 of Reference 5 are the block diagrams of the longitudinal, lateral, and directional control systems, respectively. Chapter 16 of Reference 5 contains descriptions of system operation, control law algorithms, and digital filter specifications. Since there have been several versions of the control laws to date, it should be noted that the version used in this program is OPV 8.2.1, current as of 31 August 1982.

Program modeling began by reducing the control system block diagrams (Figures 16.1, 16.2, 16.3 of Reference 5) to the forms desired for research. This was done by applying the assumptions listed in Chapter II and in Reference 1. Next, scheme of labeling the control system paths was developed. Readers of Reference 1 should note that the labeling system used there has been changed in this program to accommodate the more general nature of the full three-axis model. The paths which are modeled are:

1. Pilot inputs: pitch, roll, and yaw
2. Angle-of-attack feedback
3. Rate gyro feedbacks: pitch, roll, and yaw

4. Normal accelerometer feedback
5. Lateral accelerometer feedback
6. Stabilators, symmetrical and differential
7. Ailerons
8. Rudders
9. Leading-edge flaps, symmetrical and differential
10. Trailing-edge flaps, symmetrical and differential

The labels and nomenclature for each path are listed below:

PILOT INPUTS

PP pilot pitch path

PR pilot roll path

PY pilot yaw path

ANGLE-OF-ATTACK FEEDBACK

AA angle-of-attack path

ALPHAT computed angle-of-attack

RATE GYRO FEEDBACKS

PG pitch rate gyro path

RG roll rate gyro path

YG yaw rate gyro path

CSAOAT approx. cos(ALPHAT)

SNAOAT approx. sin(ALPHAT)

P2A,B,C filter P2 arguments

Y3A,B,C filter Y3 arguments

NORMAL ACCELEROMETER FEEDBACK

NZ normal accelerometer path

NZA nz for gain schedule use

NZAF	filtered incremental nz
NZAR1	nza-based roll rate gyro parameter
P5A,B,C	filter P5 arguments

LATERAL ACCELEROMETER FEEDBACK

NY	lateral accelerometer path

STABILATOR PATH

ST	main stabilator path
DS	differential stabilator path
ST2R	main to differential stabilator path signal
RST	right stabilator path
RSTDEF	right stabilator deflection
LST	left stabilator path
LSTDEF	left stabilator deflection

AILERON PATH

AL	main aileron path
YV1R	rudder to roll crossfeed path
RAL	right aileron path
RALDEF	right aileron deflection
LAL	left aileron path
LALDEF	left aileron deflection

RUDDER PATH

RD	main rudder path
RSR	roll surface to rudder interconnect path
LRD	left rudder path
LRDDEF	left rudder deflection
Y5A,B,C	filter Y5 arguments

LEADING-EDGE FLAP PATH

LE	main leading-edge flap path
DLE	differential leading-edge flap path
RLE	right leading-edge flap path
RLEDEF	right leading-edge flap deflection
LLE	left leading-edge flap path
LLEDEF	left leading-edge flap deflection

TRAILING-EDGE FLAP PATH

TE	main trailing-edge flap path
DTE	differential trailing edge-flap path
RTE	right trailing-edge flap path
RTEDEF	right trailing-edge flap deflection
LTE	left trailing-edge flap path
LTEDEF	left trailing-edge flap deflection

Nomenclature from the McDonnell Aircraft Company literature [Refs. 5 and 6] has been included where it was deemed helpful for reference. This primarily includes such terms as PK_, RK_, YK_, PV_, RV_, and YV_; these are constants or variables which occur at significant points. The terms QC (dynamic pressure) and PS (static pressure) are also from the McDonnell Douglas Aircraft Company literature. Carter [Ref. 1] lists and defines some additional terms which have been used in this program, including filter and constant nomenclature. The remainder of the terminology was developed specifically for this program. Block diagrams incorporating the simplying assumptions mentioned

previously and using the above notation are included in Appendix A. These particular block diagrams are intended only to aid in identifying the various terms with their location in the diagrams and leave out much of the detail found in Figures 16.1, .2, .3 of Reference 5.

Carter [Ref. 1, Chapter III] gives an excellent description of the methodology of using CSMP to model the block diagram elements of Figures 16.1, 16.2, and 16.3 of Reference 5. Some changes have been made to those procedures, however:

1. The control law gain schedules, referred to singularly as Function F__, have been removed from nosort sections. Instead, all of the gain schedules have been coded as Fortran functions and called as required. This step has removed over 300 lines of code from the CSMP program, since Fortran subroutines or functions do not count against the allowable number of CSMP statements. Twenty-four functions have been added for the lateral and directional axes, in addition to 13 for the longitudinal axis discussed in Reference 1:

FUNCTION	DESCRIPTION
F4	Roll Rate Feedback Gain Schedule (QC,PS)
F6	Differential Stabilator Gain Schedule (RI,PS,RV11)
F7	Lateral Command Gain Schedule (QC,PS)
F10	Rudder Command Gain (QC)
F13	Lateral Command Gain Schedule (RI,PS,STORES)

F17	Rudder Pedal Command Gain Increment (AOA)
F30	RSRI Gain Schedule (QC,PS)
F31	Differential T.E. Flap Gain Schedule (RI,PS)
F34	Differential T.E. Flap Gain Schedule (AOA)
F35	Lateral Forward Loop Gain Schedule (AOA)
F36	Aileron Gain Schedule (QC,PS,RI)
F38	RSRI Gain Schedule (AOA,RI,PS)
F39	Rudder-Roll Interconnect Gain Schedule (AOA)
F41	Roll Surface Limit Schedule (AOA,RI)
F42	RSRI Nonlinear Gradient
F45	Directional Forward Loop Gain Schedule (QC,PS)
F90	Lateral Acceleration Feedback Gain Schedule (RI,PS)
F93	Differential L.E. Flap Gain Schedule (RI,PS,NZAF)
F96	Yaw Rate Gain Schedule (QC,PS)
F101	Differential Stabilator Load Alleviation (RI,PS, NZAF)
F108	Directional Inertial Gain Schedule (QC)
F112	Lateral Acceleration Gain (RI)
F113	Lateral Acceleration Gain (AOA)
F114	Rudder Pedal Command Gain Increment (RI)

Algorithms for the implementaion of the functions are found on Chapter 16 of Reference 5.

2. Frequency averagers and rate limiters are modeled using CSMP macro statements. Thus, all nosort sections have been eliminated from the current 3-axis program.

3. Cross-axis signal paths for differential stabilators, leading-edge and trailing-edge flaps have been added. Rolling-surface-to-rudder interconnect (RSRI), rudder-to-roll crossfeed, and inter-axis distribution of rate gyro feedback signals have been incorporated.

4. As noted in Reference 1, in the longitudinal-only simulation, the stabilators and flaps were modeled for one direction of motion only; this constraint has been removed in the current 3-axis program.

Descriptions of the algorithms used to implement A/D and D/A converters, digital filters, aliasing filters, limit functions, and actuator servomechanisms are contained in Reference 1. They are used in the current program without modification.

IV. MODEL VALIDATION

The process of validating the computer model of the F/A-18 flight control system was carried out in two major steps:

1. Individual block diagram elements were tested using probable ranges of input values. Digital filter macros, rate limiters, frequency averagers, and gain schedule functions are included here. Carter [Ref. 1] describes the testing of all of the block diagram elements mentioned with the exception of the gain schedule functions relevant to the lateral and directional axes, which were individually verified by this researcher exactly as were the gain schedule functions used in Reference 1.

2. The entire model was assembled and subjected to inputs from the stick, rudder, and angle-of-attack, acceleration, and feedback sensors. The goal of this step was to verify correct direction of motion of control surfaces in response to unambigous inputs.

Inasmuch as Reference 1 has already discussed the procedures and results relevant to Step 1, with the exceptions noted, the current discussion will involve Step 2. Further, since Reference 1 contains the validation results for stabilator, leading-edge and trailing-edge flaps due to pitch inputs, here the control surface motion due to roll

and yaw inputs will be of primary concern. The source of information used as a reference in comparing model performance to design specification was the McDonnell Aircraft Company F/A-18 Flight Control System Design Report [Ref. 5].

The computer model requires the following inputs:

1. Static pressure (PS) in lb/sq.ft.
2. Dynamic pressure (QC) in lb/sq.ft.
3. Pilot pitch input (PP1) in lb. + = aft stick.
4. Pilot roll input (PR1) in lb, + = right stick.
5. Pilot yaw input (PY1) in lb, + = right rudder.
6. Pitch rate gyro feedback (PG1) in deg/sec, + = nose-up.
7. Roll rate gyro feedback (RG1) in deg/sec, + = right roll.
8. Yaw rate gyro feedback (YG1) in deg/sec, + = right yaw.
9. Angle-of-attack (AA1) in deg.
10. Normal acceleration (NZ1) in "g", + = nose-up motion.
11. Lateral acceleration (NY1) in "g", + = nose-right motion.

Depending on the need, inputs can be programmed in CSMP as step or ramp functions (see Reference 4), or using any standard Fortran function such as SIN or EXP, for example. Combinations thereof are acceptable, as well. An example

of a test roll input consisting of a 6lb right stick input for two seconds followed by a 6lb left stick input as written in CSMP would be:

PR1 = 6.0*STEP(0.0) - 12.0*STEP(2.0)

For this purposes of the tests described in this thesis, PS and QC are held constant through test maneuvers, though there exists no such constraint in the model. The effects of varying PS and QC were verified to be correct when each gain schedule function was compared to the graphical data in Chapter 16 of Reference 5. Rate gyro feedback, angle-of-attack, and acceleration inputs may currently be programmed as described for pilot inputs. However, accurate representation of these latter three types of inputs will not be possible until the F/A-18 aerodynamic build-up by Raithel [Ref. 2] and the non-linear equations of motion are incorporated into the program. As that time, those inputs will be determined by the program and will not be explicity stated. For the tests described here, rate gyro feedback, angle-of-attack, and acceleration data, when necessary to simulate aircraft responses, will be provided by "best guess" estimation of those data, given the pilot input.

Figures presented in this chapter are based on a time scale of four seconds per maneuver. Control surface deflections are in degrees. Static pressure and dynamic

pressure inputs are listed on each figure in terms of altitude and Mach number. Standard day conditions are assumed in all cases. Angle-of-attack is occasionally listed where it plays a significant role in shaping the control system response.

Figure 4.1. depicts differential stabilator response to a +/- 6.0lb lateral stick input. The shift from right to left stick takes place at the 2.0 second mark. The pertinent gain schedule functions are Functions 6 and 101. For the given flight conditions and pilot input, the stabilator response is very close to maximum. The response is decreased at higher dynamic pressures.

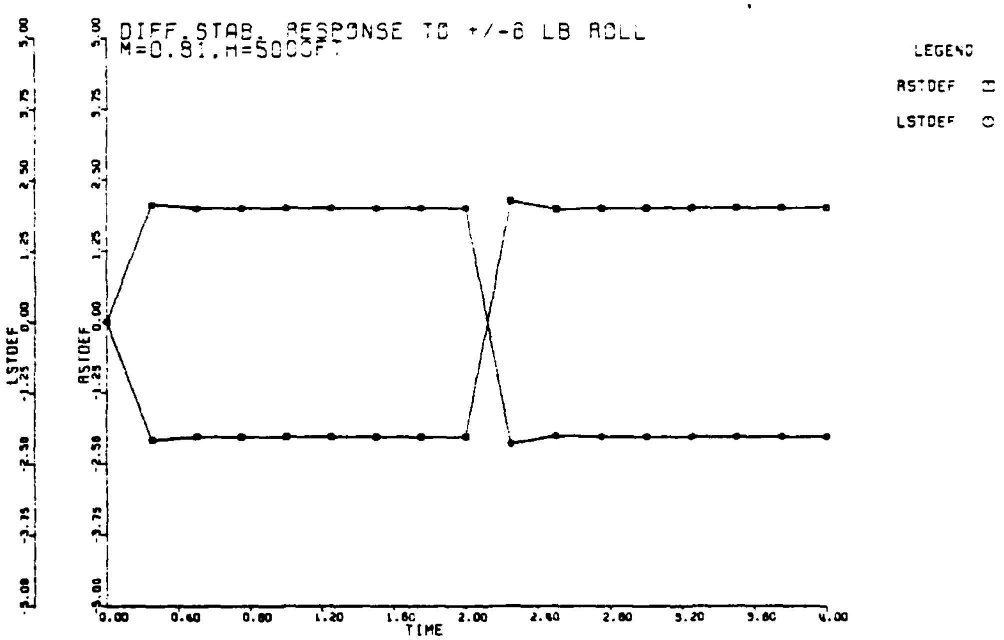


Figure 4.1. Differential Stabilator Response to Roll

Figure 4.2. shows differential stabilator response to the same input as for Figure 4.1., but with roll rate gyro feedback added as a ramp type input. This causes the damping effect on the stabilator motion. The overshoot caused when the input signals are reversed in sign is to some extent believed to be an artifact of the programming of the roll rate feedback input, which only approximates an actual feedback input. This phenomena can be noticed on some of the following graphs, as well.

Figure 4.3. shows differential stabilator motion due to a rudder input. This demonstrates the rudder-to-roll surface CAS interconnect, which is gain-scheduled by Function 39 using angle-of-attack. A ramp input was used to simulate the angle-of-attack signal.

Aileron motion due to a +/- 6.0lb roll input is depicted in Figure 4.4. Aileron gain schedules are Functions 35 and 36, which accept angle-of-attack and air data inputs, respectively. Both functions decrease gain as their input values increase: in the case of increasing angle-of-attack, this is to reduce sideslip; the air data gain schedule lessens aileron response at high dynamic pressures where aeroelastic effects would otherwise cause aileron reversal.

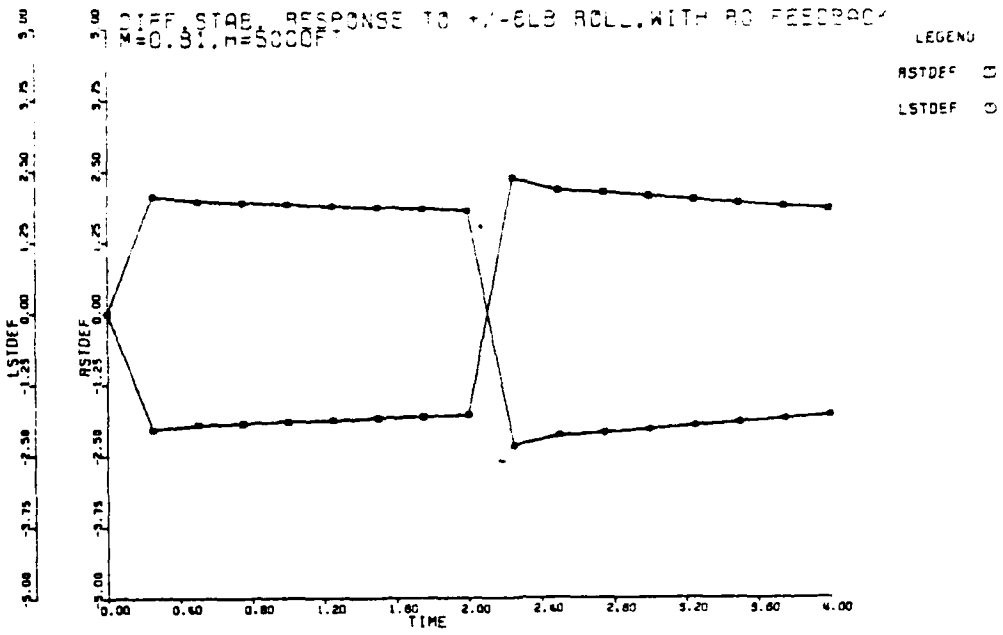


Figure 4.2. Differential Stabilator Response with Roll Rate Feedback

When roll rate feedback is added to the aileron roll motion,it is seen that a damping effect is present, as it should be. This is shown in Figure 4.5. Roll rate feedback is gain-scheduled by Function 4, which acts to decrease feedback gain as dynamic pressure increases. The overall lateral control system gain is shaped additionally by Functions 7 and 13, using air data inputs. These gain-schedules do not easily lend themselves to straight-forward explanations of purpose, indeed, there are certainly multiple purposes for the design of these functions. Reference 5 is more explicit as concerns the gain-schedule design philosophy.

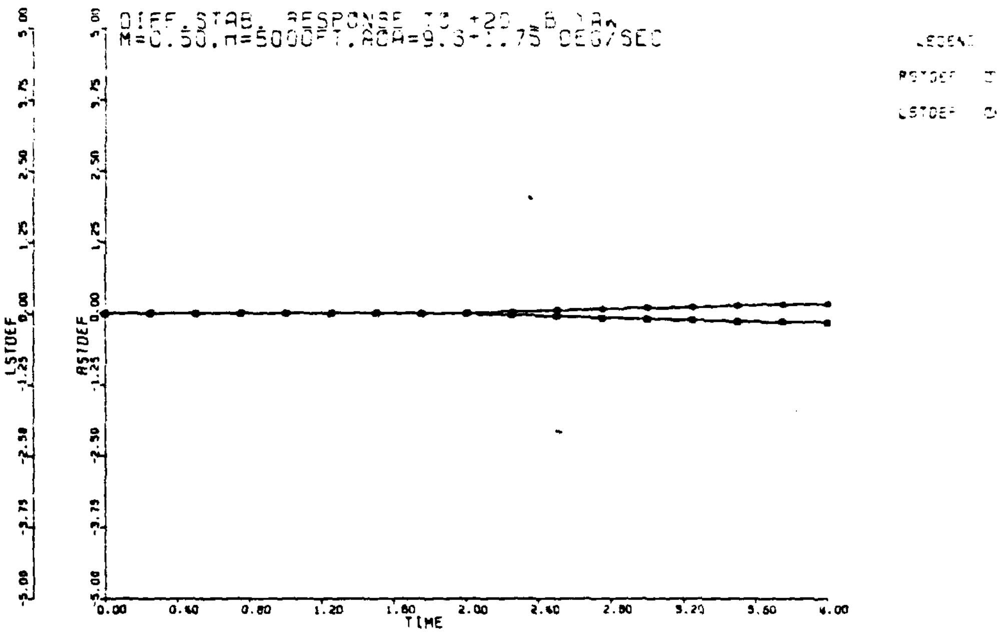


Figure 4.3. Differential Stabilator Response to Yaw

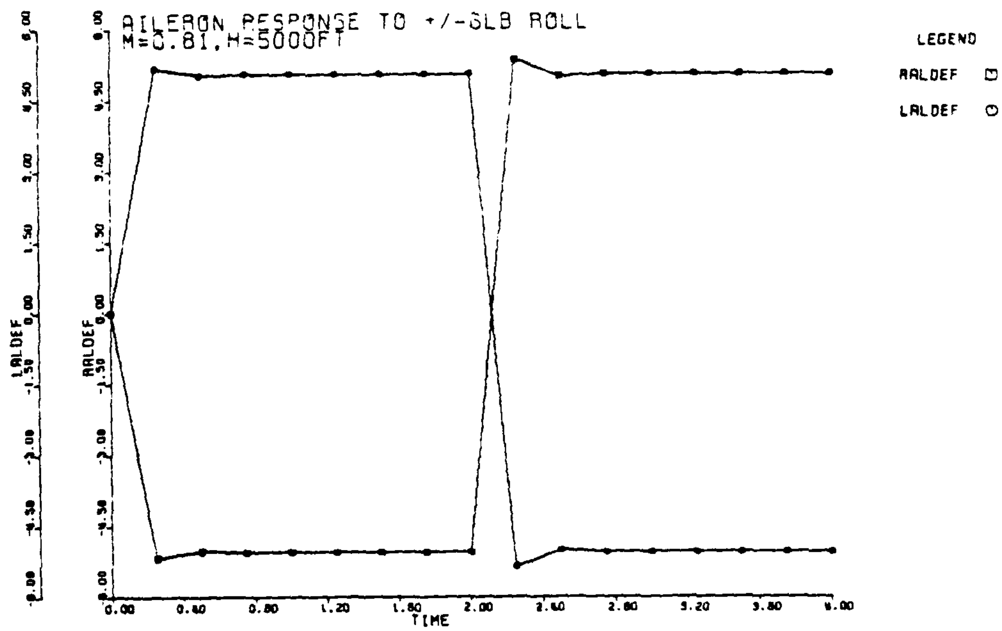


Figure 4.4. Aileron Response to Roll

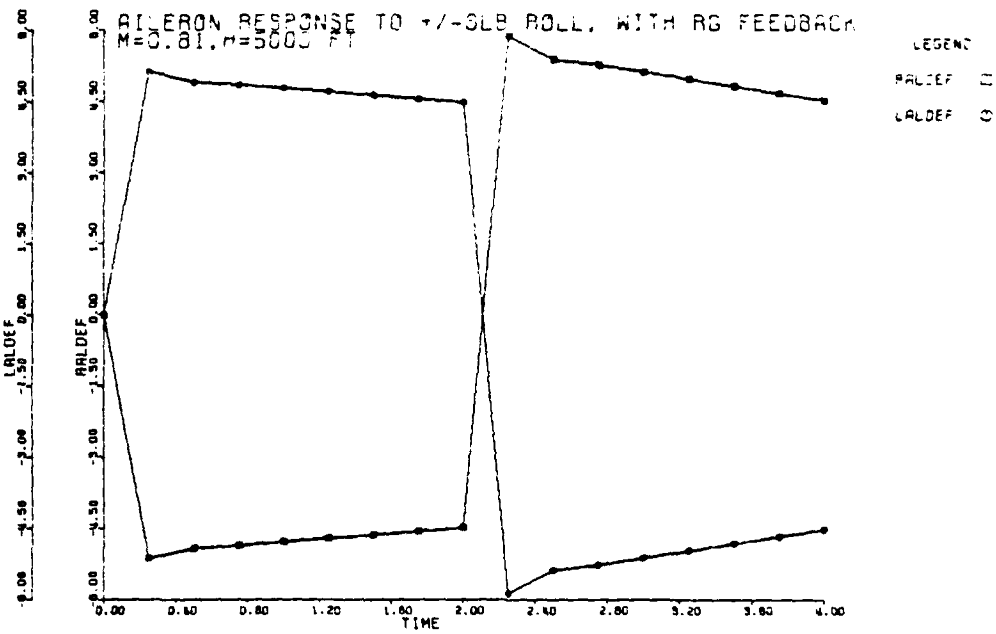


Figure 4.5. Aileron Response with Roll Rate Feedback

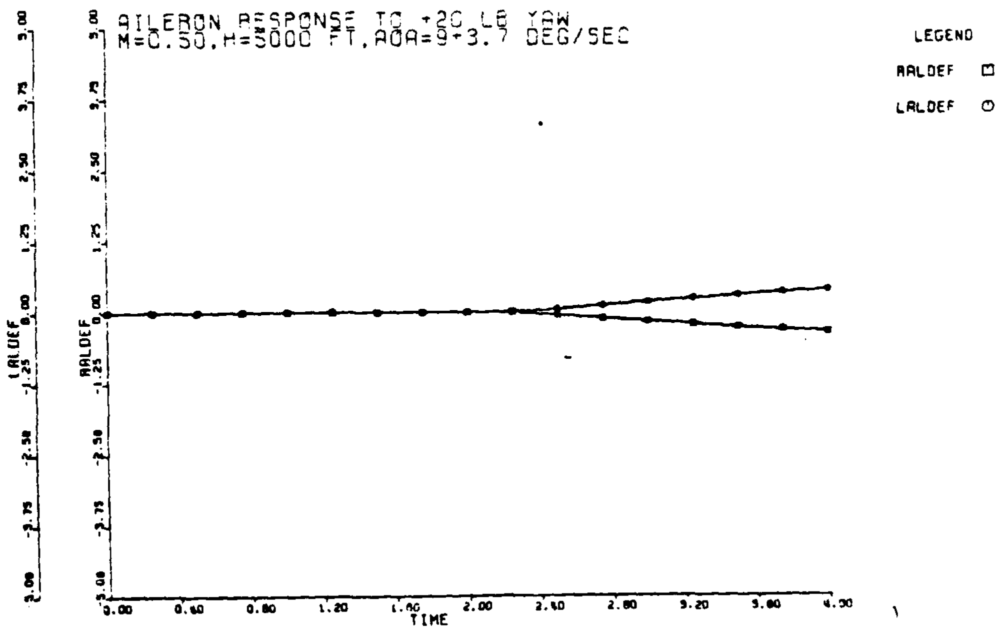


Figure 4.6. Aileron Response to Yaw

The rudder-to-roll surface CAS interconnect signal causes an aileron deflection as angle-of-attack is increased. This effect is shown in Figure 4.6., and is analogous to that noted in Figure 4.3.

At higher dynamic pressures, as noted earlier, aileron gain is decreased to reduce aeroelastic effects. Differential leading-edge and trailing-edge flaps are incorporated to maintain an acceptable roll response under these circumstances. Differential leading-edge flap motion in response to a +/- 10.0 lb roll input is displayed in Figure 4.7. At the indicated Mach number, aileron response is nil. Function 93 governs the response as a function of air data and normal acceleration inputs.

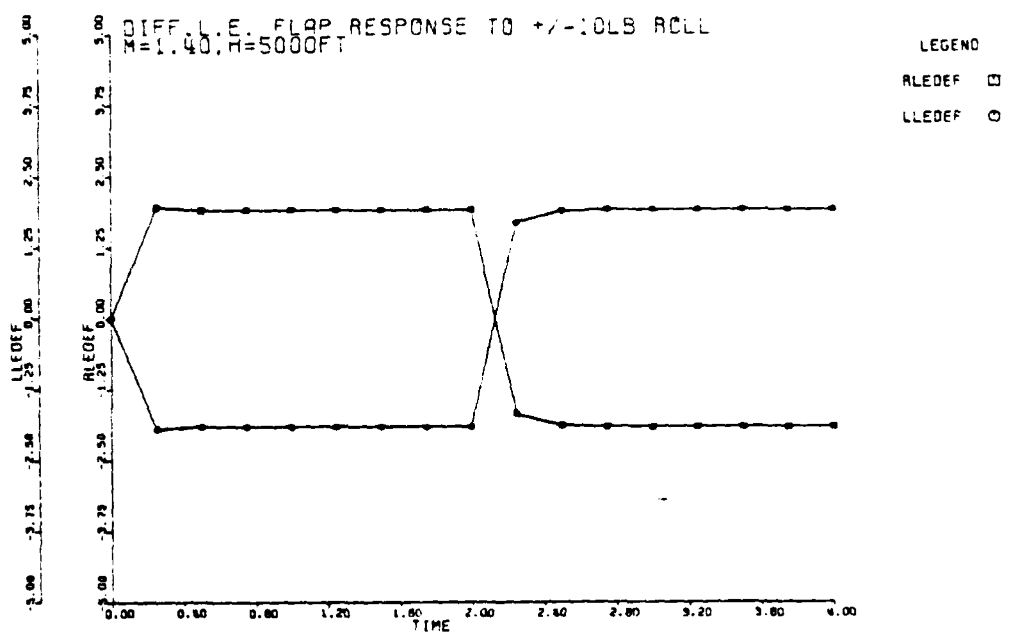


Figure 4.7. Differential Leading-Edge Flap Response to Roll

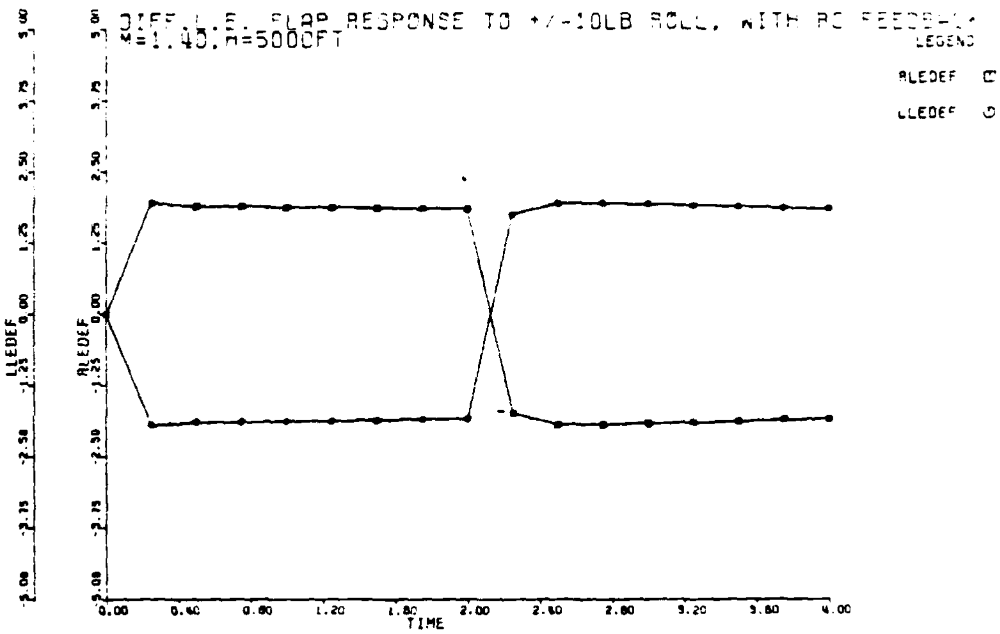


Figure 4.8. Differential Leading-Edge Flap Response to Roll with Roll Rate Feedback

The expected effect of roll rate feedback on differential leading-edge flap motion is shown in Figure 4.8. The damping is noticeably less than for the differential stabilator and aileron damping observed previously; this is due to the inherently higher damping of aircraft rolling motion at the dynamic pressures where differential flap motion is required. Figures 4.9. and 4.10. depict similar responses for differential trailing-edge flaps, without and with roll rate feedback, respectively. Functions 31 (air data) and 34 (angle-of-attack) are the gain schedules responsible for differential trailing-edge flap motion.

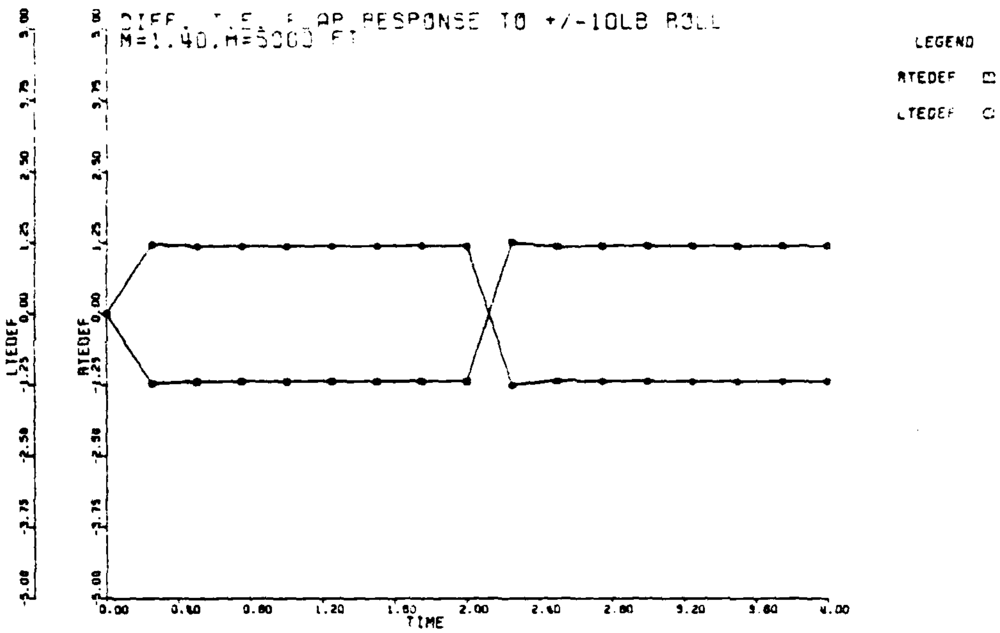


Figure 4.9. Differential Trailing-Edge Flap Response to Roll

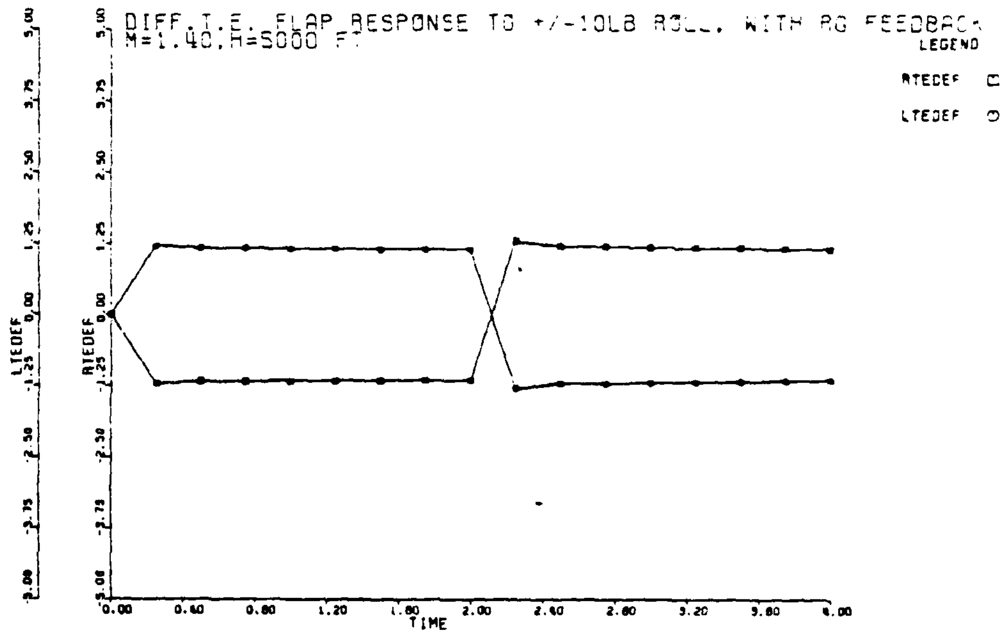


Figure 4.10. Differential Trailing-Edge Flap Response to Roll with Roll Rate Feedback

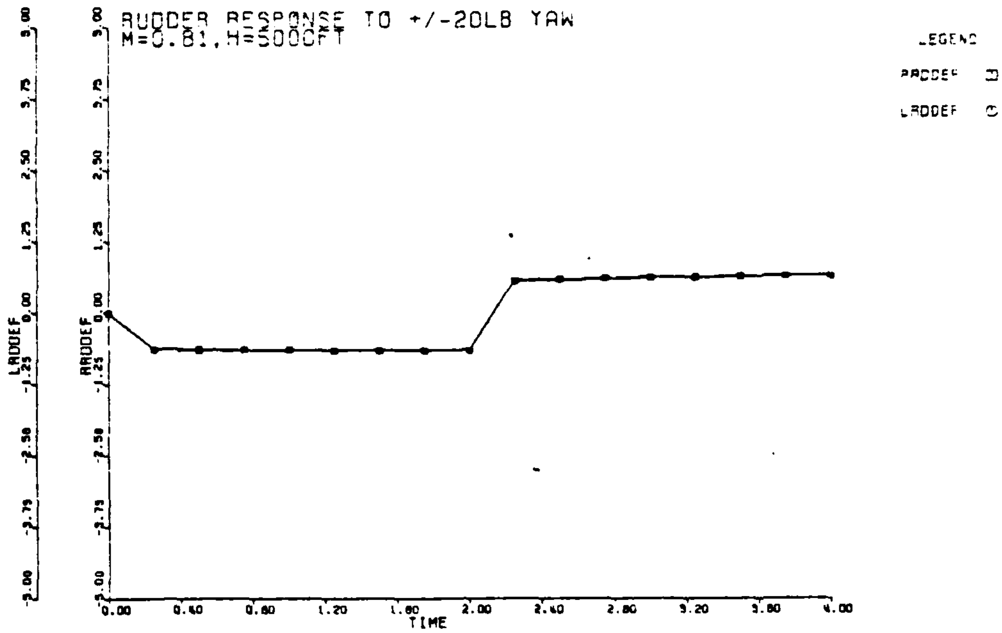


Figure 4.11. Rudder Response to Yaw

Directional control system responses are made by the twin rudders, which do not act differentially in the Auto Flap Up mode. Rudder response to a +/- 20.0lb rudder pedal input is shown in Figure 4.11. The gain schedules which shape the motion are Functions 10 (air data), 71 (air data), and 114 (angle-of-attack). Figure 4.12. shows rudder motion when feedback signals are added to the simulation. The damping effect is the sum of stability axis yaw rate (this feedback signal is a blend of yaw and roll rates, and angle-of-attack data), and lateral acceleration feedbacks.

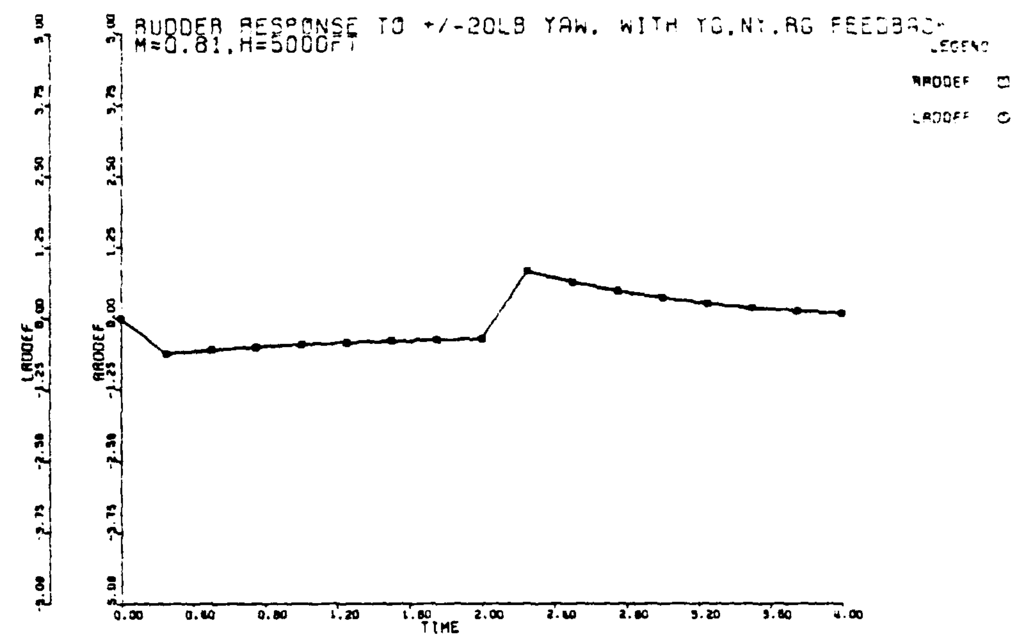


Figure 4.12. Rudder Response to Yaw with Feedback

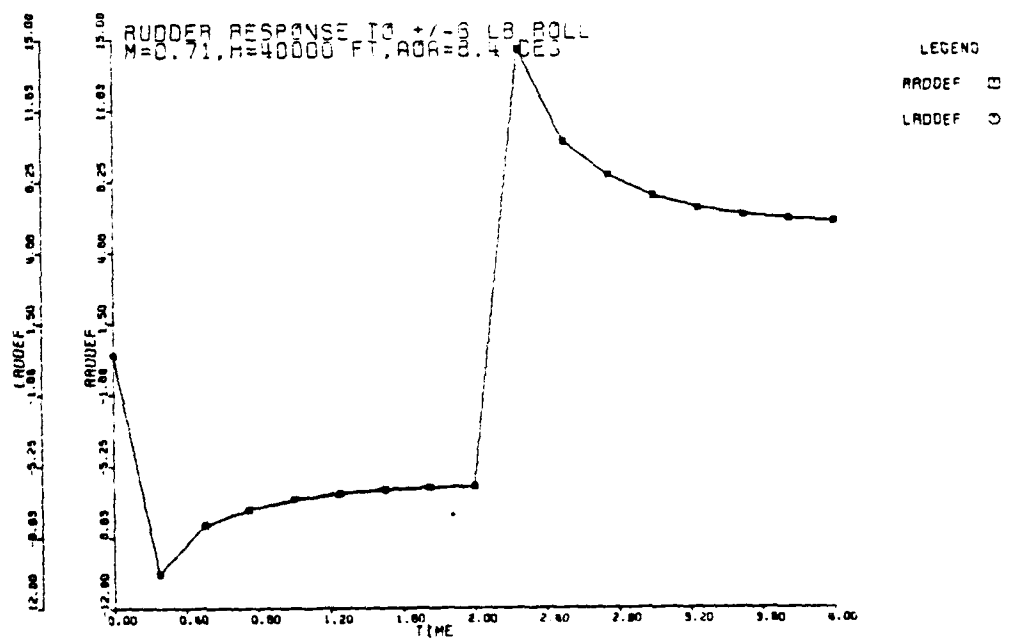


Figure 4.13. Rudder Response to Roll

Several gain schedules are involved here in shaping rudder response to the requirements of sideslip reduction, prevention of excessive vertical tail loads, inertia coupling reduction, and high angle-of-attack maneuvering stability. Reference 5 discusses these considerations.

Rolling surface-to-rudder interconnect signal operation can be seen in Figure 4.13. Functions 30 (air data), and 38 (angle-of-attack and air data) shape this response.

Verification of all of the signal paths of the F/A-18 flight control system in the Auto Flap Up mode has thus been accomplished. The gain schedules (Functions) were each individually verified prior to incorporation into the computer model.

The process of model validation has thus been to observe model responses to some relatively basic inputs, and then to judge these responses as being plausible or not with respect to the data provided in Reference 5. The computer model has given logical results as noted in this chapter, and also for a wider range of input signals and flight conditions than have been discussed here. The conclusion is that the flight control system model is valid representation of the F/A-18 flight control system within the range of the simplifying assumptions and limitations noted previously in Reference 1 and in this thesis.

V. CONCLUSIONS AND RECOMMENDATIONS

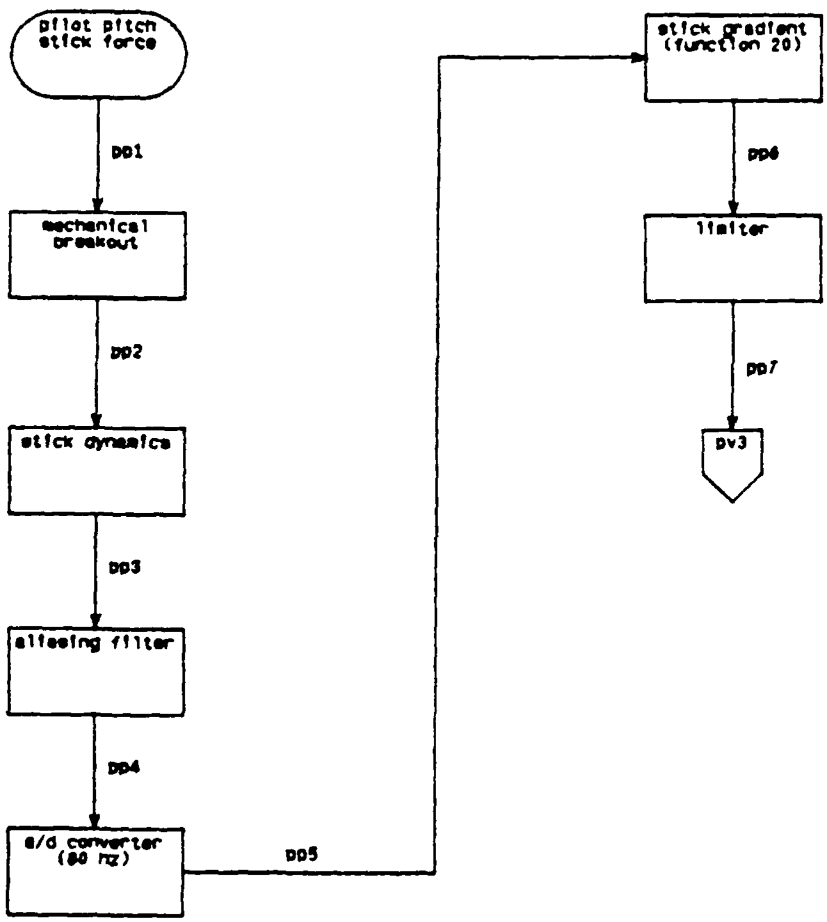
The computer simulation of the F/A-18 digital electronic flight control system developed here is suitable for further use in aircraft control systems studies. With the inclusion of F/A-18 aerodynamics and equations of motion, it will also serve to examine simulated maneuvering flight under conditions found near or at the accepted flight envelope. Comparative studies may be done of newer control systems concepts, using the known response as the basis for comparison.

CSMP has size limitations for several internal parameters, the most pertinent of which are total number of program statements (1900), and maximum number of statements in any given sort section (600). The current program is right at the latter limit with 599 statements in one sort section. Some statements could be combined to reduce this number; this is not recommended until the user is familiar with the F/A-18 flight control system and its representation here. It is recommended that the aerodynamics and equations of motion be written as Fortran subroutines and called at the beginning of the dynamic section. This would use only two lines, thus only two additional statements would have to be combined to make the necessary room. There are not obvious dividing lines where the current program could be

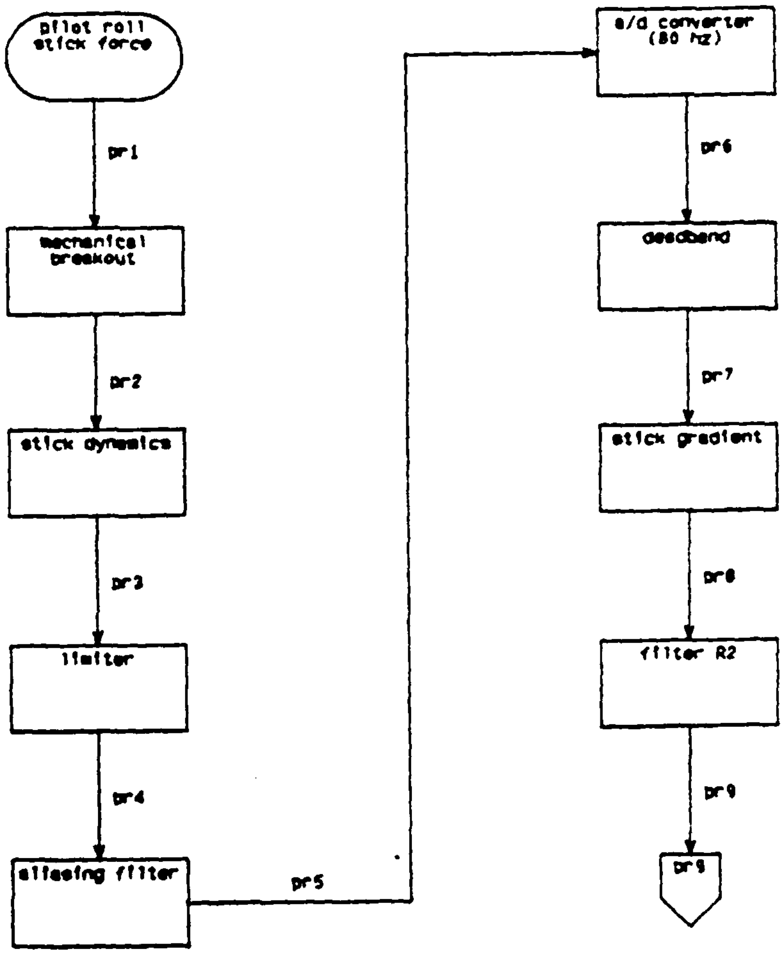
broken up into more than one sort section, however, users experienced in CSMP may find this to be possible when the aerodynamics and equations of motion are incorporated.

APPENDIX A: MODEL BLOCK DIAGRAMS

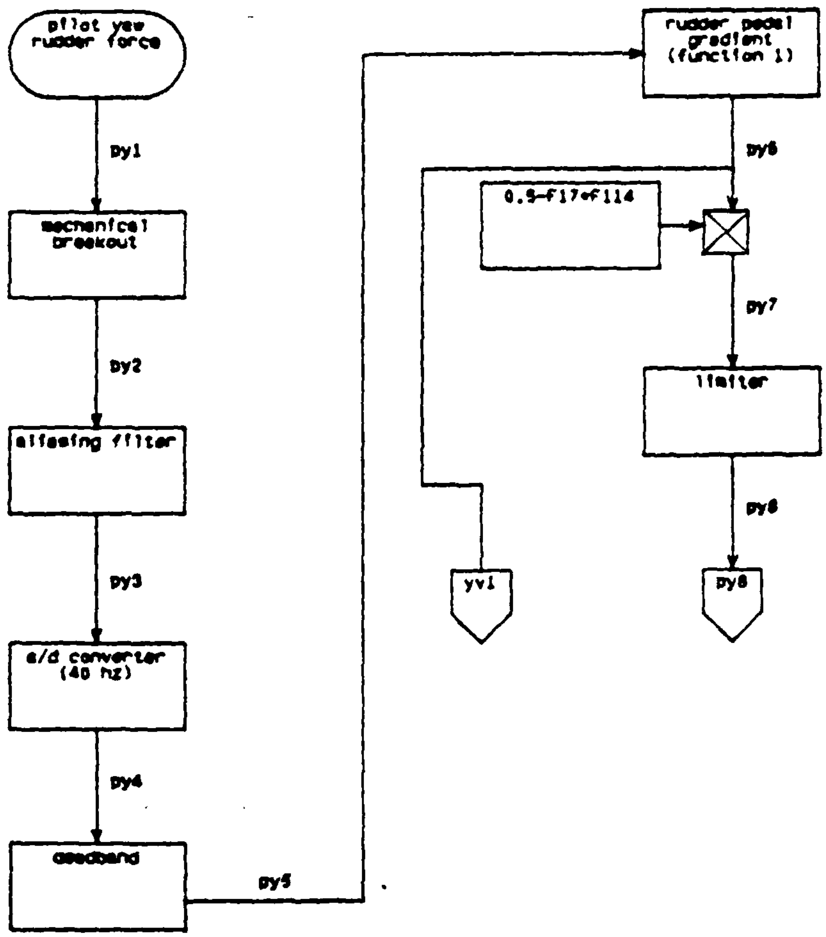
PILOT PITCH INPUT PATH



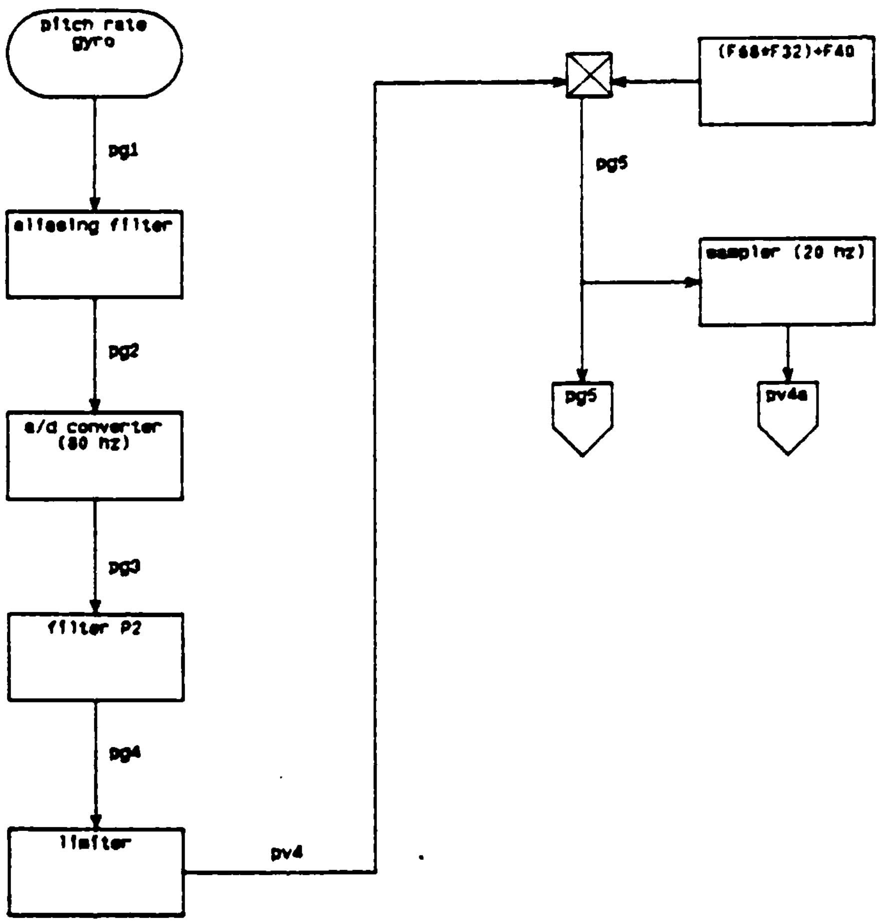
PILOT ROLL INPUT PATH



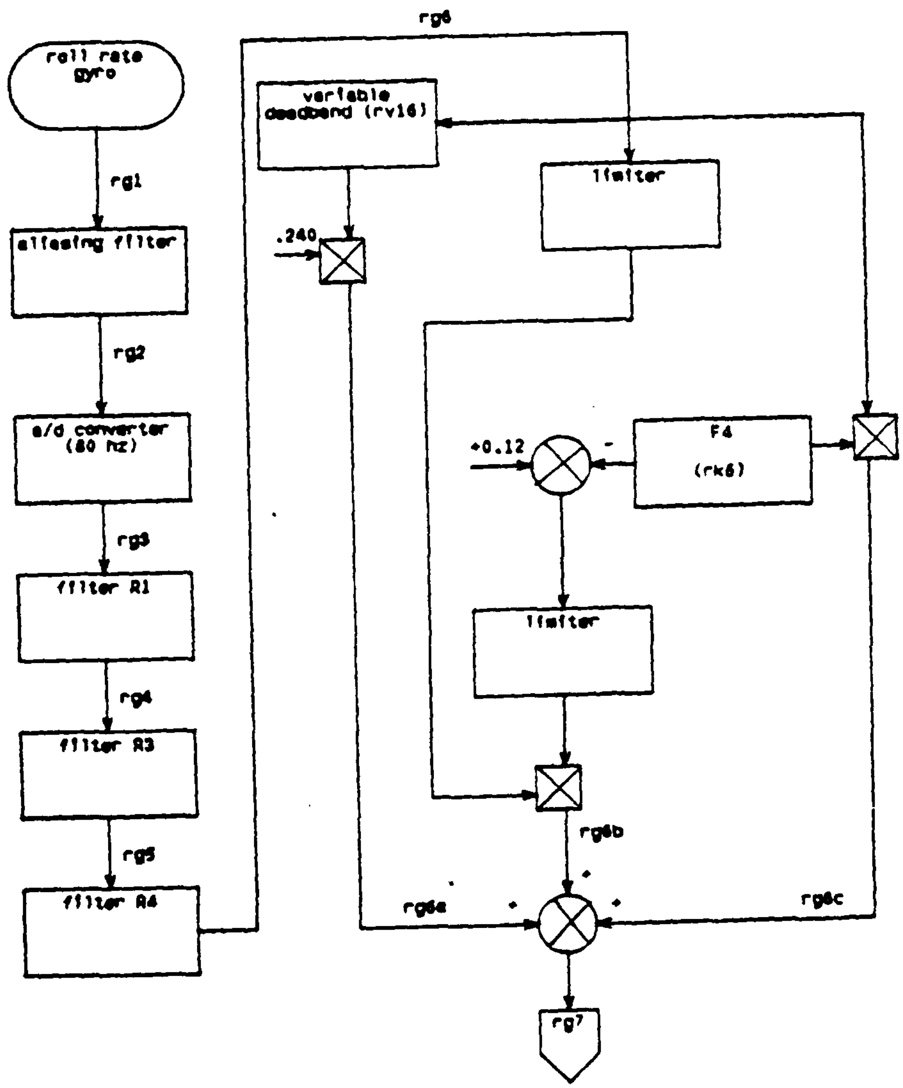
PILOT YAW INPUT PATH



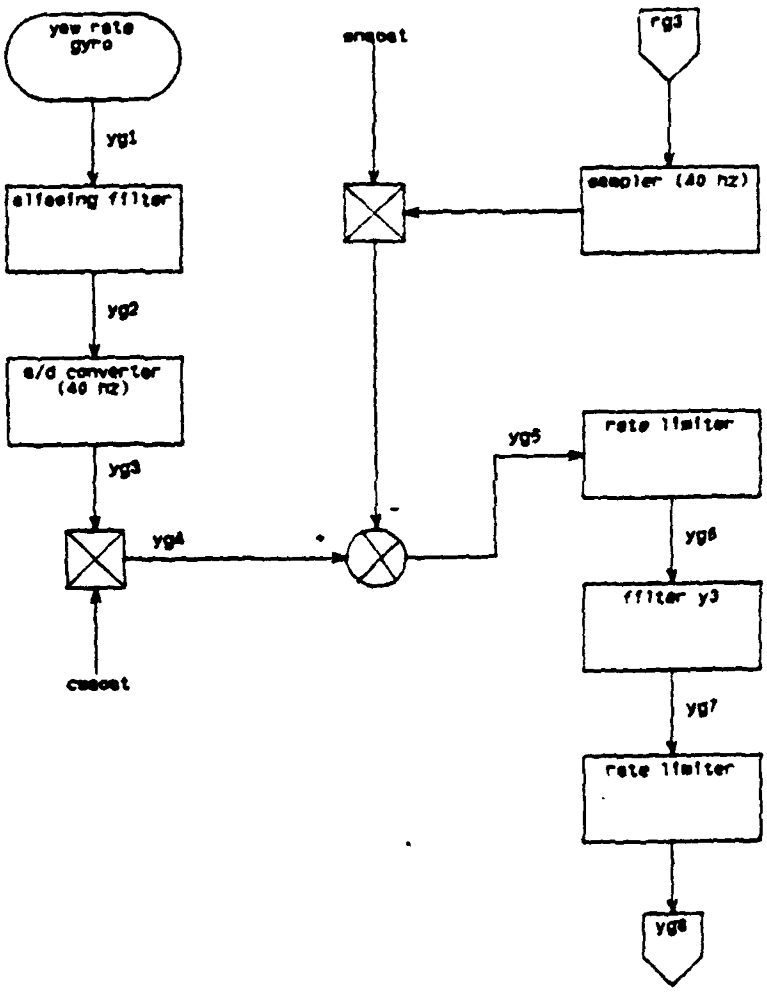
PITCH RATE GYRO PATH



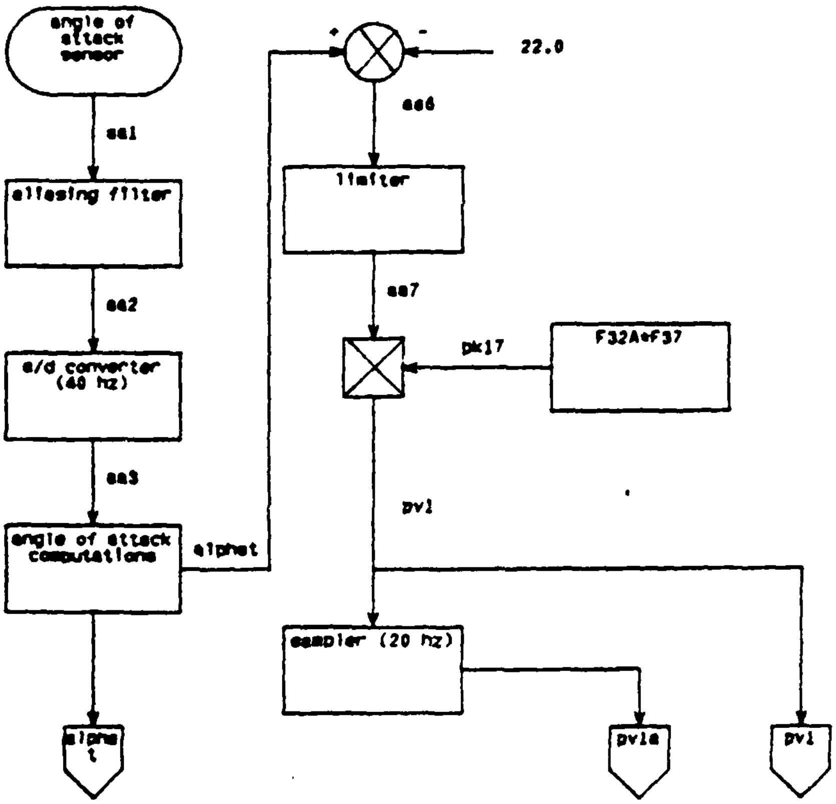
ROLL RATE GYRO PATH



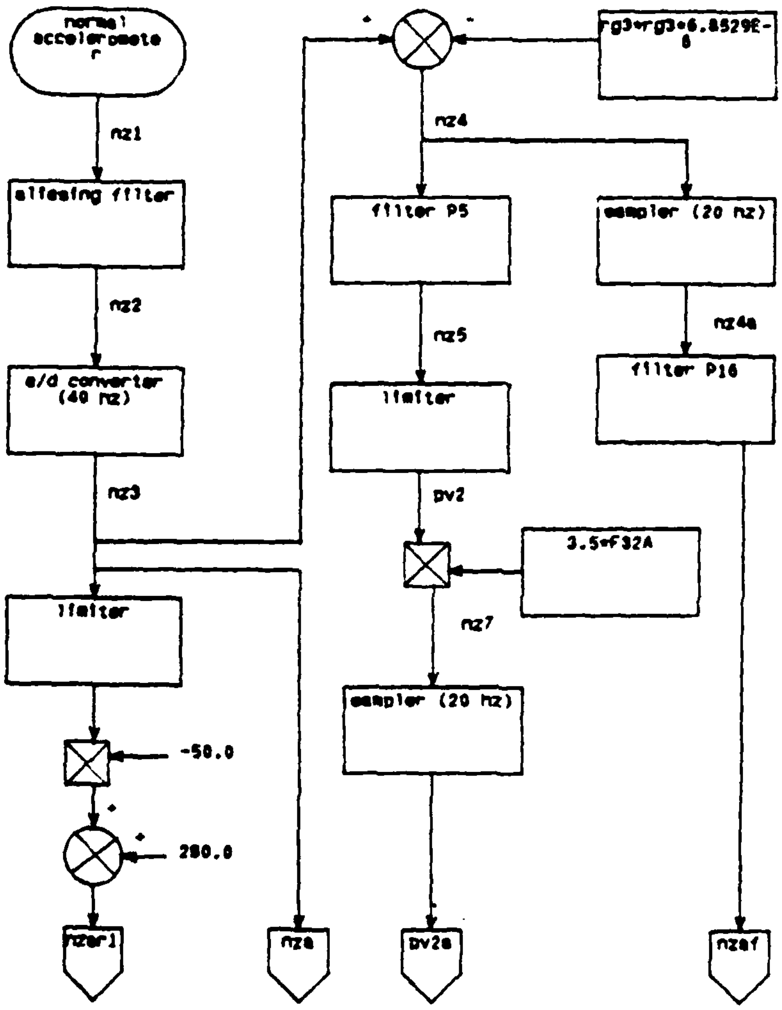
YAW RATE GYRO PATH



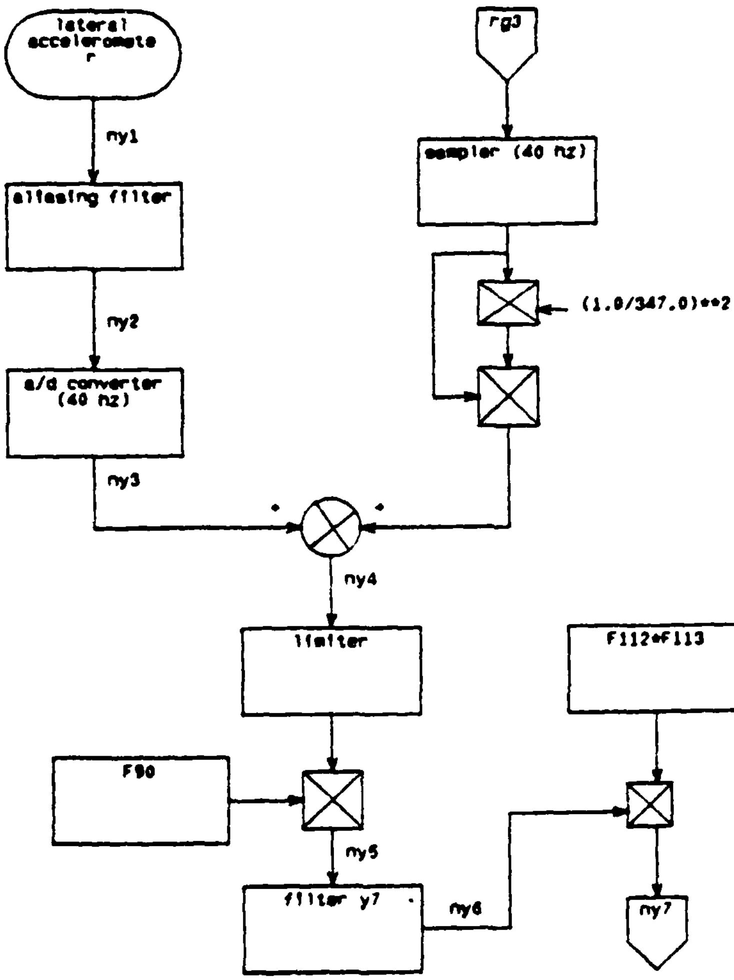
ANGLE-OF-ATTACK SENSOR PATH



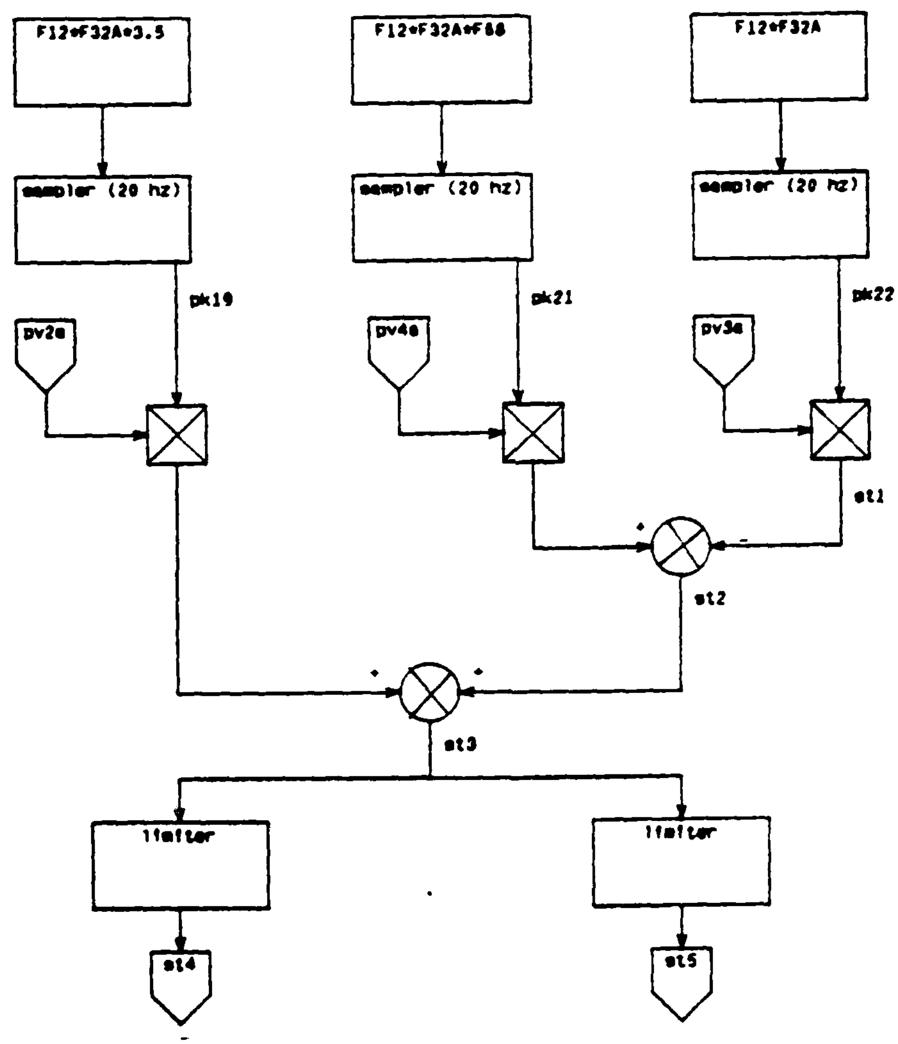
NORMAL ACCELEROMETER SENSOR PATH



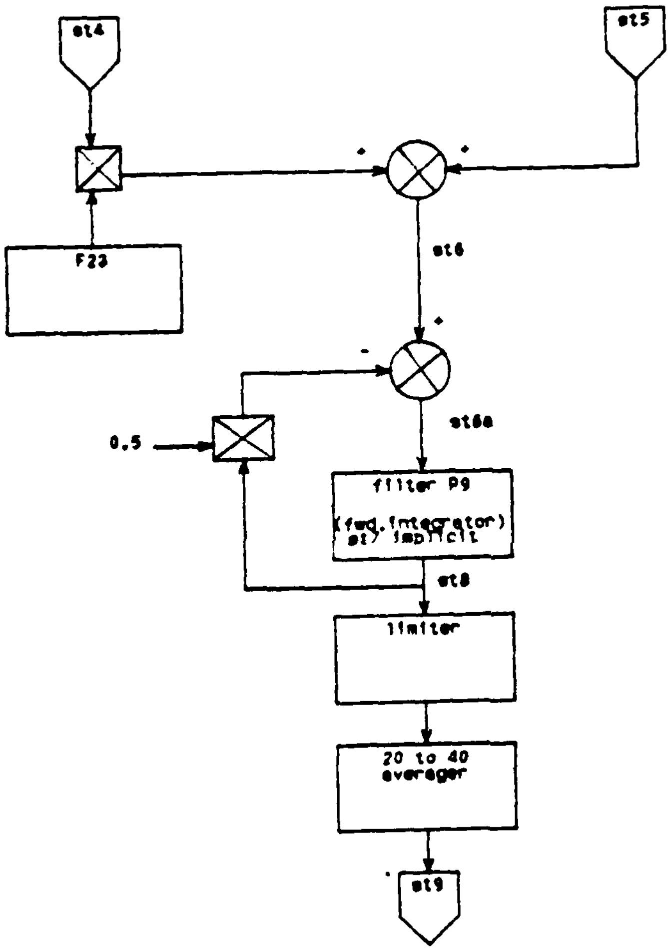
LATERAL ACCELEROMETER SENSOR PATH



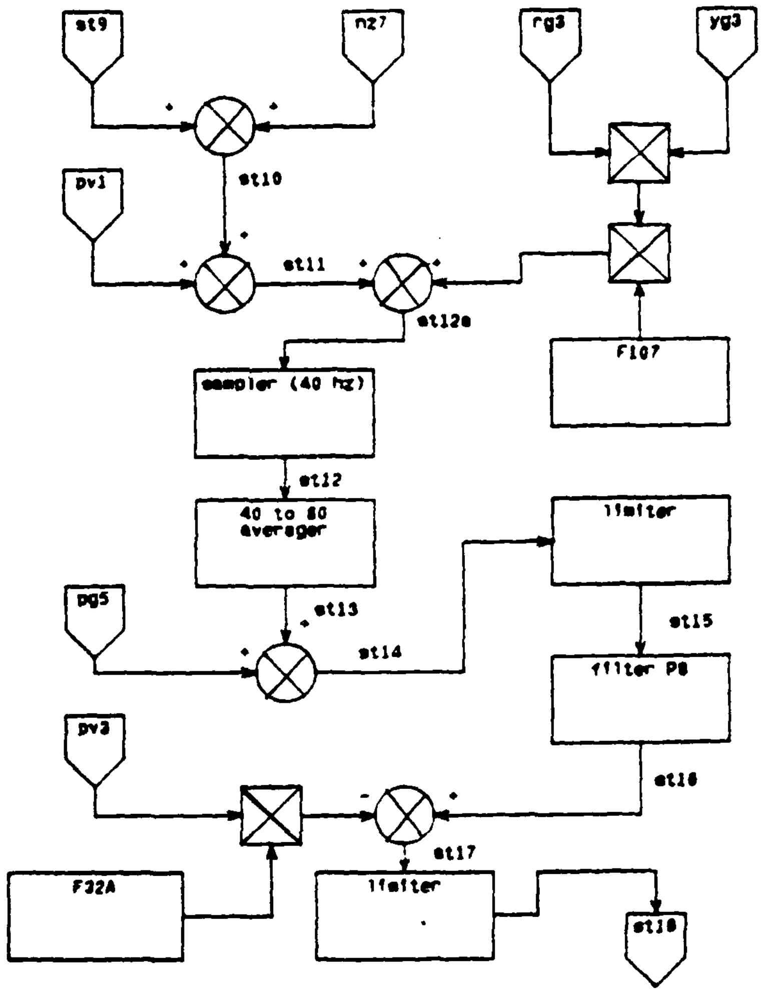
STABILATOR PATH



STABILATOR PATH (CONT.)

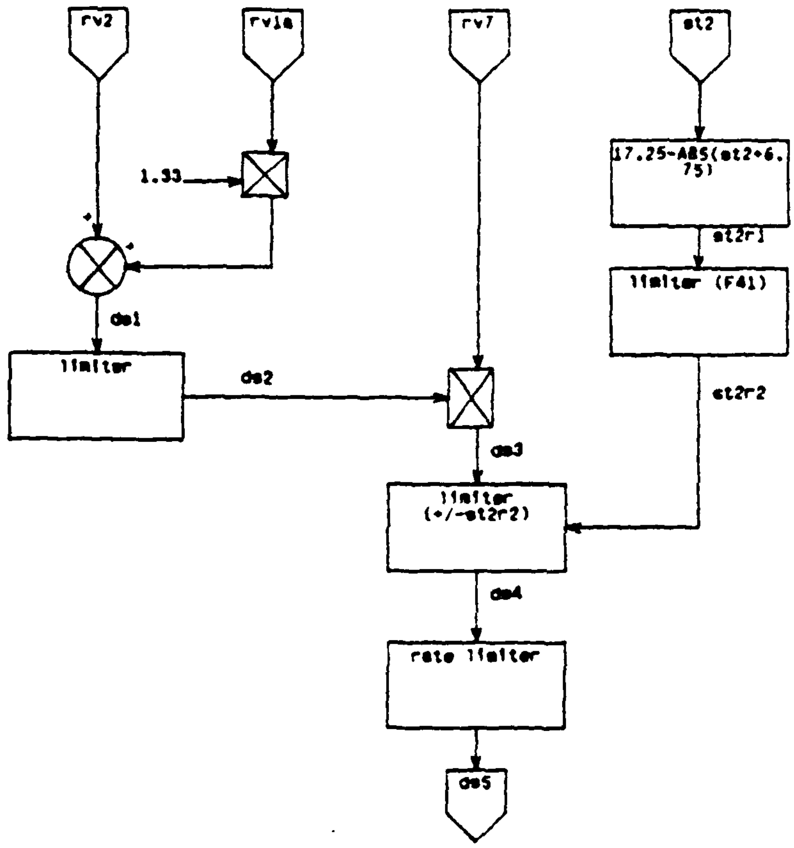


STABILATOR PATH (CONT.)

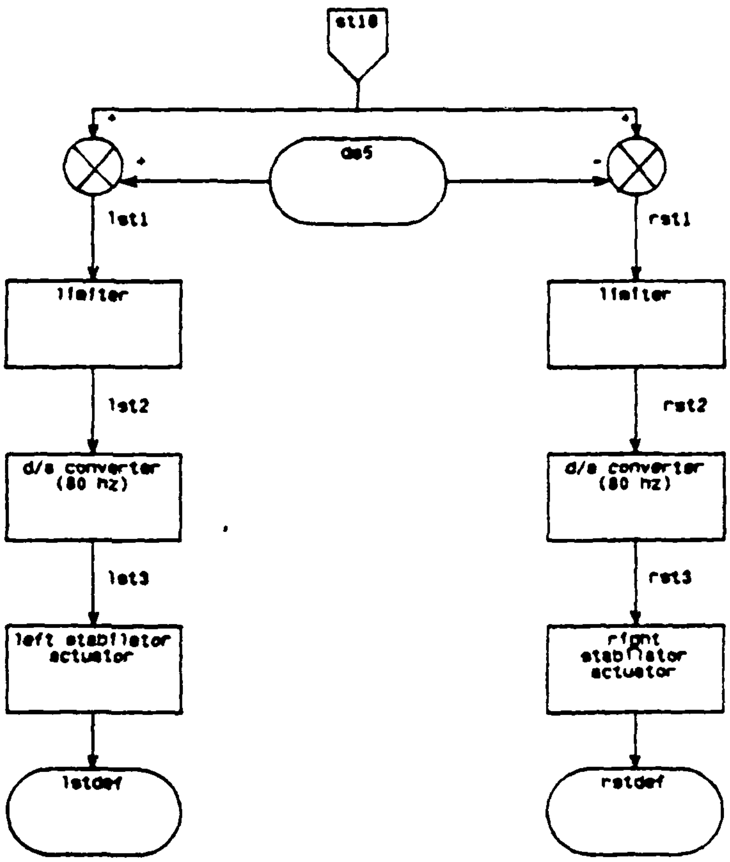


STABILATOR PATH (CONT.)

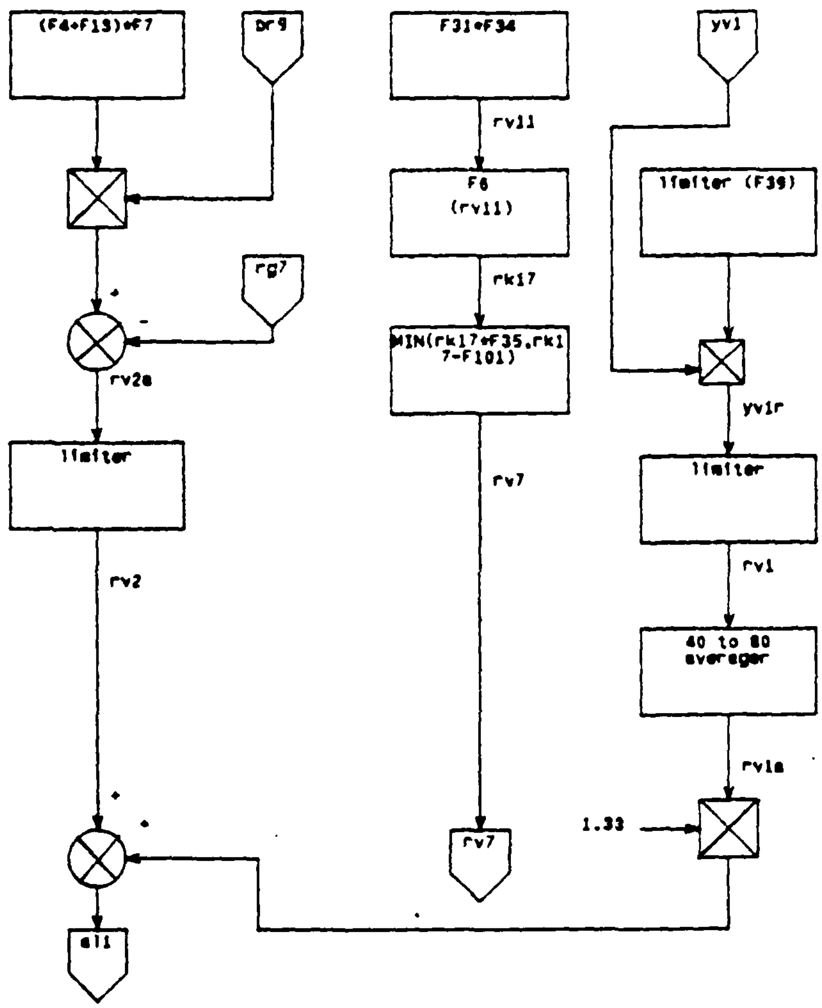
DIFFERENTIAL STABILATOR SIGNAL PATH



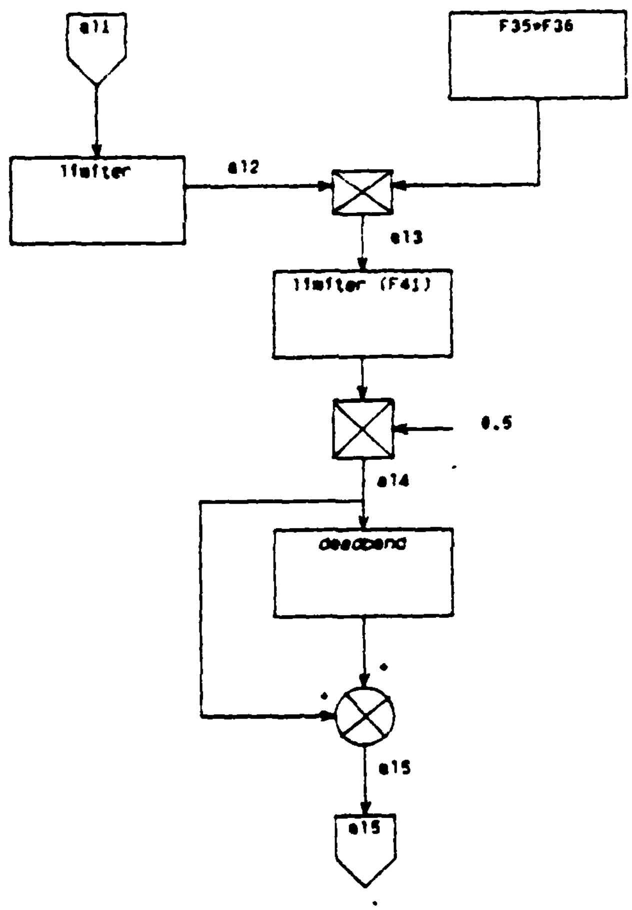
STABILATOR PATH (CONT.)



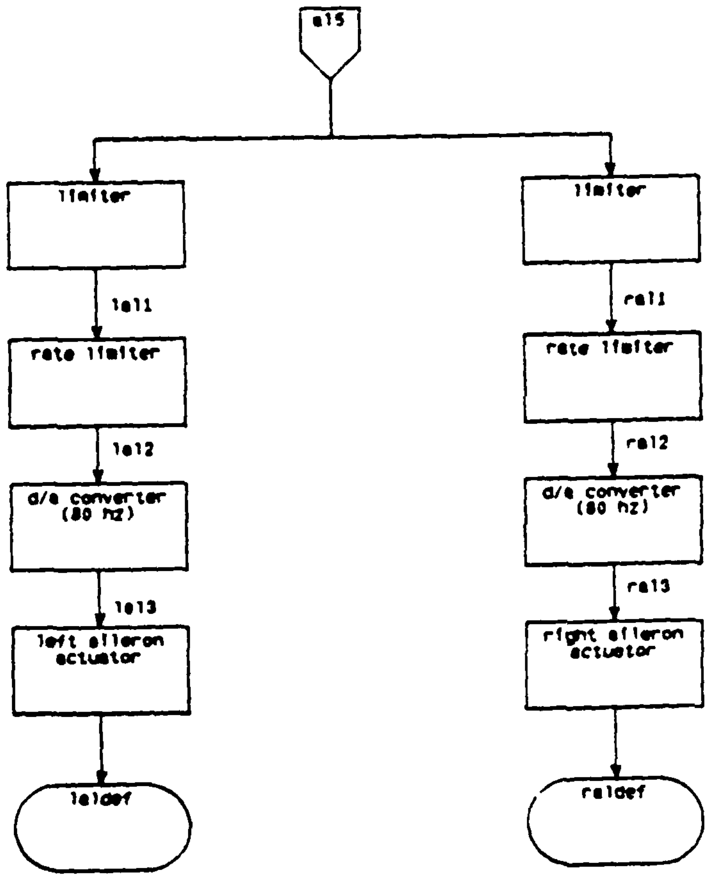
AILERON PATH



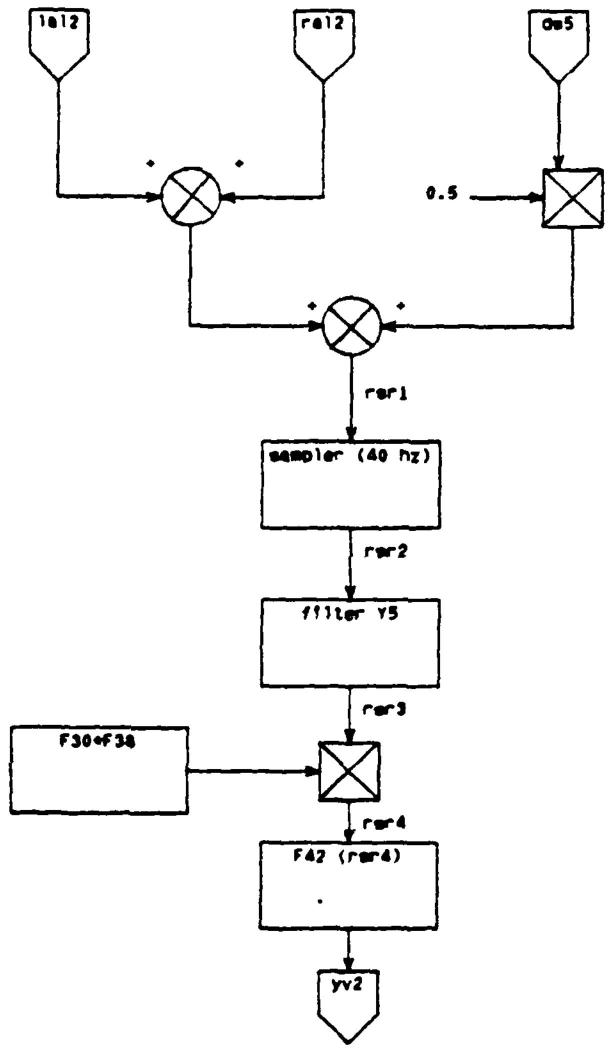
AILERON PATH (CONT.)



AILERON PATH (CONT.)

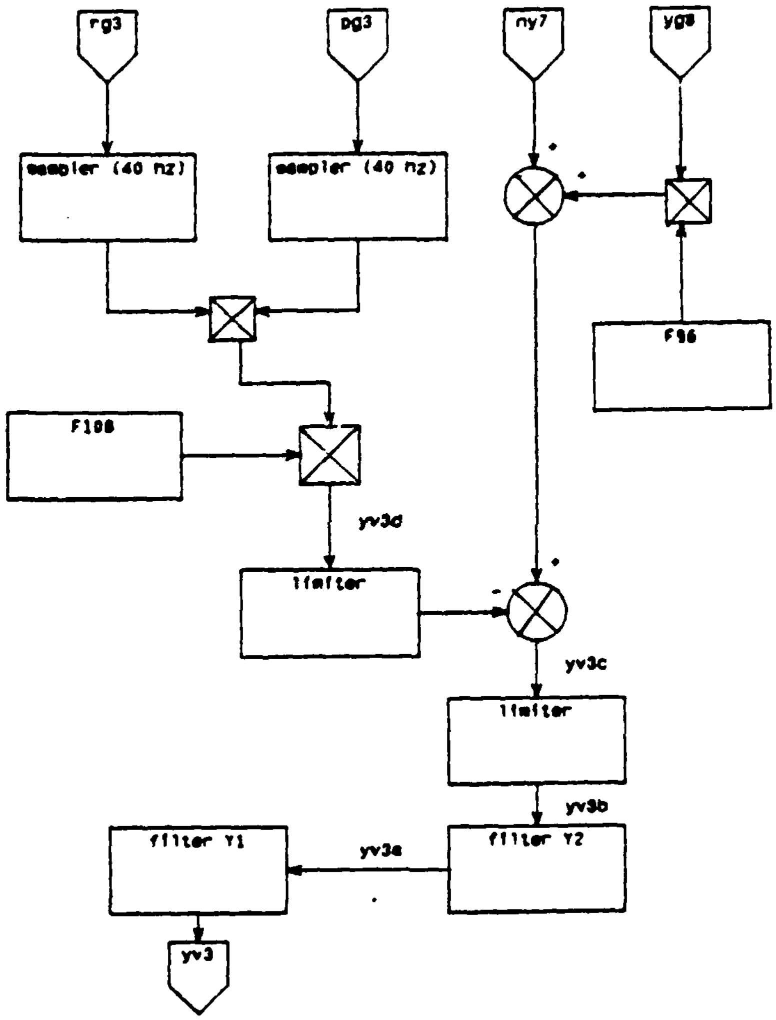


RUDDER PATH

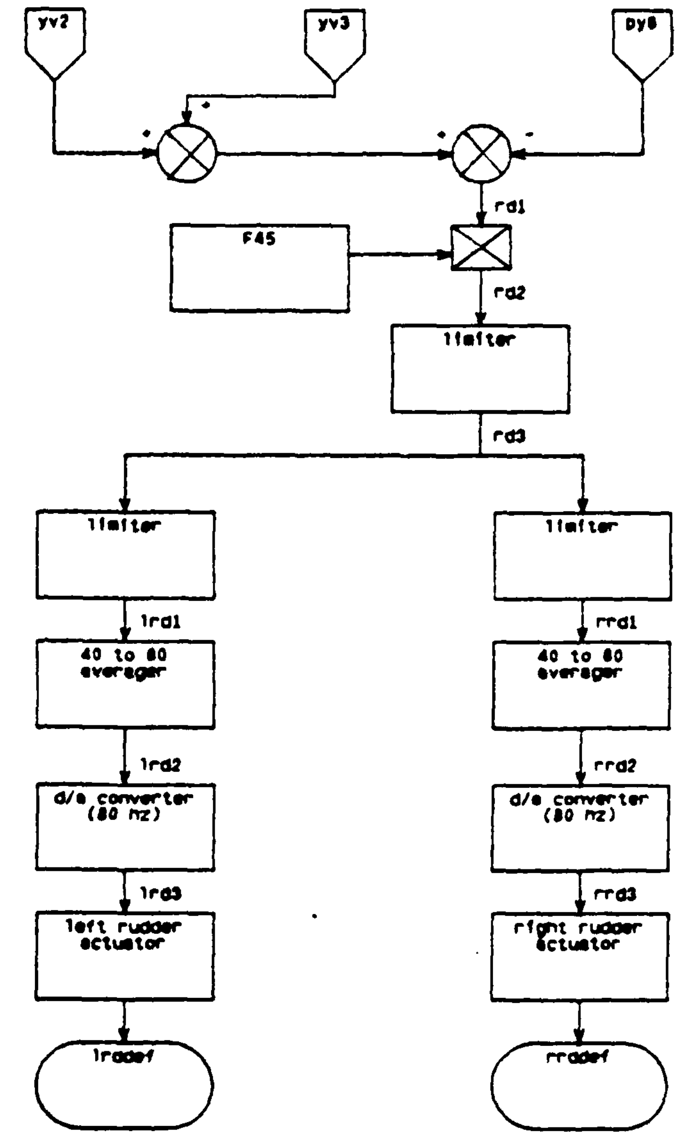


lal2
ral2
Ow5
+
+
0.5
+
+
rsr1
sampler (40 hz)
rsr2
filter Y5
rsr3
F30+F38
rsr4
F42 (rsr4)
yv2

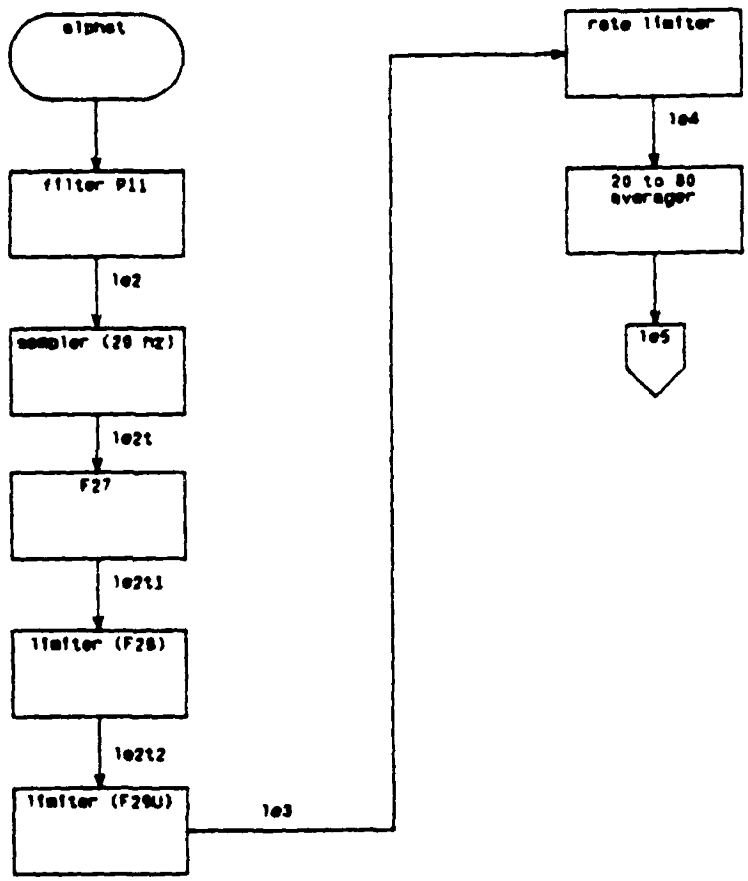
RUDDER PATH (CONT.)



RUDDER PATH (CONT.)

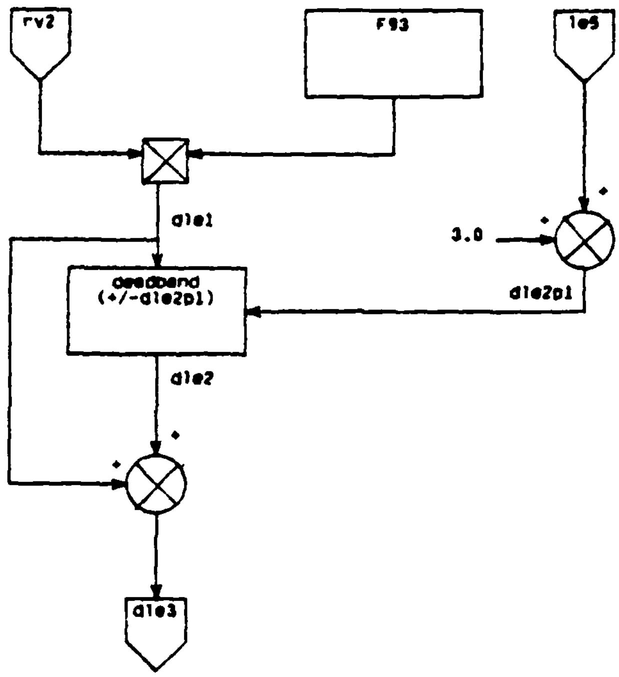


LEADING-EDGE FLAP PATH

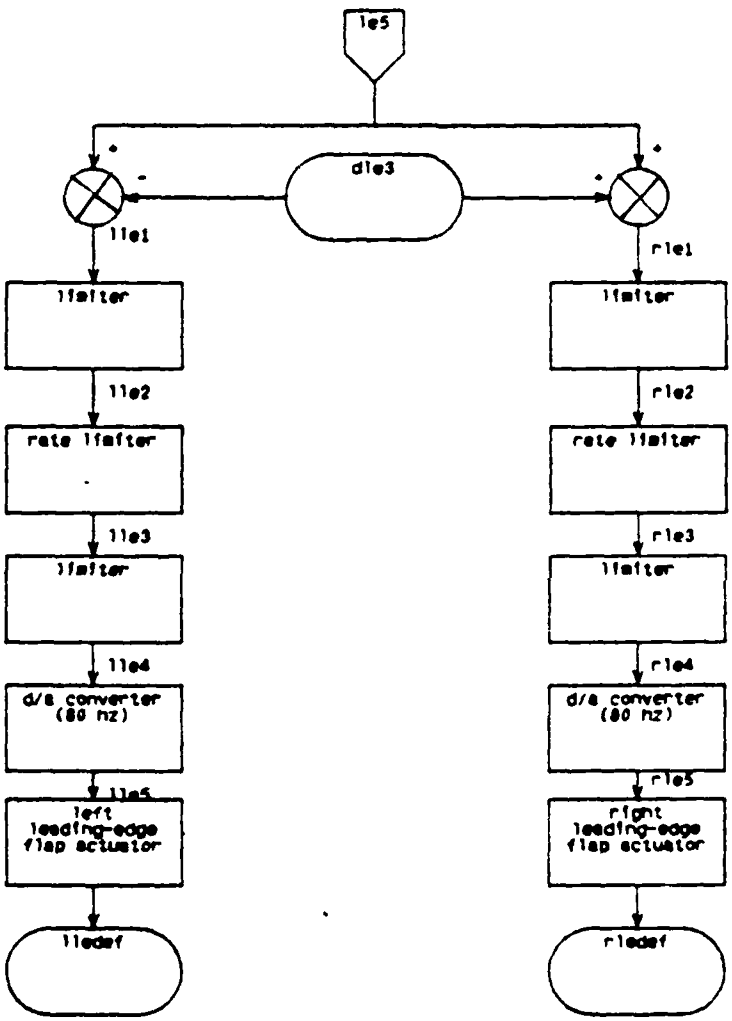


LEADING-EDGE FLAP PATH (CONT.)

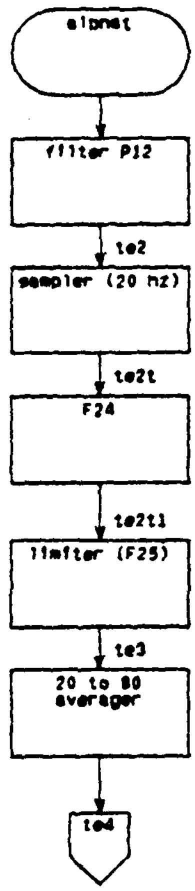
DIFFERENTIAL LEADING-EDGE FLAP PATH



LEADING-EDGE FLAP PATH (CONT.)

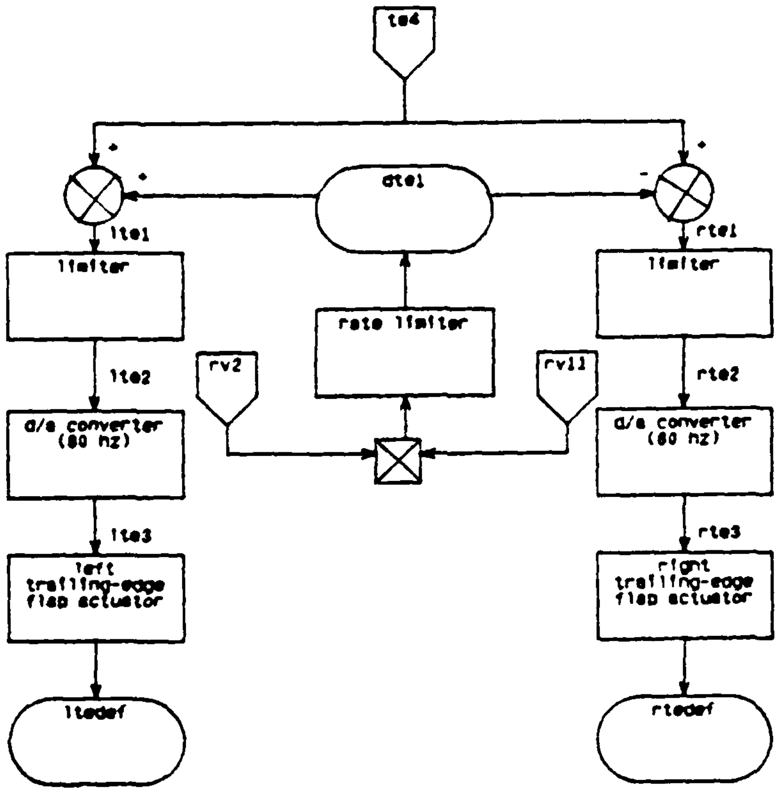


TRAILING-EDGE FLAP PATH



TRAILING-EDGE FLAP PATH (CONT.)

DIFFERENTIAL PATH INCLUDED



APPENDIX B: MODEL COMPUTER PROGRAM

```
*        ************************************************
*        **                                            **
*        **      F-18  FLIGHT CONTROL LAW SIMULATION   **
*        **                                            **
*        ************************************************
*
*      ***************************************************
*      *****INTEGRATING,LAG,AND NOTCH DIGITAL FILTERS*****
*      ***************************************************
*
MACRO FOUT=ZINT(FIN,KA,KB,IMP,FOUTZ1)
    PROCEDURAL
    IF(IMP.NE. 1.0) GO TO 10
    IF(KEEP.NE.1.0) GO TO 10
    IF(TIME.EQ.0.0) GO TO 10
    FOUT=KA*FIN + KB*FOUTZ1
    FOUTZ1=FOUT
10  CONTINUE
ENDMAC
MACRO FOUT=ZLAG(FIN,KA,KB,KC,IMP,FINZ1,FOUTZ1)
    PROCEDURAL
    IF(IMP.NE. 1.0) GO TO 10
    IF(KEEP.NE.1.0) GO TO 10
    IF(TIME.EQ.0.0) GO TO 10
    FOUT=KA*FIN - KB*FINZ1 + KC*FOUTZ1
    FOUTZ1=FOUT
    FINZ1=FIN
10  CONTINUE
ENDMAC
MACRO FOUT=ZNOTCH(FIN,KA,KB,KC,KD,KE,IMP,FINZ1,FINZ2,FOUTZ1,FOUTZ2)
    PROCEDURAL
    IF(IMP.NE. 1.0) GO TO 10
    IF(KEEP.NE.1.0) GO TO 10
    IF(TIME.EQ.0.0) GO TO 10
    FOUT=KA*FIN + KB*FINZ1 + KC*FINZ2 - KD*FOUTZ1 - KE*FOUTZ2
    FOUTZ2 = FOUTZ1
    FOUTZ1 = FOUT
    FINZ2 = FINZ1
    FINZ1 = FIN
10  CONTINUE
ENDMAC
*
*       ****************************
*       *****FREQUENCY AVERAGERS*****
*       ****************************
*
MACRO  Z40=AV2040(Z20,IMP)
     PROCEDURAL
        IF (KEEP.NE.1.0) GO TO 20
        IF (TIME.NE.0.0) GO TO 5
        Z20Z1=Z20
        Z40Z1=Z20
        DEL=0.0
        GO TO 10
5       IF (IMP.EQ.1.0) GO TO 10
        Z40=Z40Z1+DEL
        GO TO 15
10      Z40=Z20
        DEL=(Z20-Z20Z1)/2.0
        Z20Z1=Z20
        Z40Z1=Z40
15      CONTINUE
20      CONTINUE
ENDMAC
MACRO  Z80=AV4080(Z40,IMP)
     PROCEDURAL
        IF (KEEP.NE.1.0) GO TO 20
        IF (TIME.NE.0.0) GO TO 5
        Z40Z1=Z40
        Z80Z1=Z40
        DEL=0.0
        GO TO 10
5       IF (IMP.EQ.1.0) GO TO 10
        Z80=Z80Z1+DEL
        GO TO 15
10      Z80=Z40
```

```
      DEL=(Z4C-Z40Z1)/2.0
      Z40Z1=Z40
      Z8CZ1=Z8C
15    CONTINUE
20    CONTINUE
ENDMAC
MACRO Z80=AV208C(Z2C,IMP)
    PROCEDURAL
      IF (KEEP.NE.1.C) GO TO 20
      IF (TIME.NE.0.0) GO TO 5
      Z2CZ1=Z2C
      Z80Z1=Z20
      GO TO 10
5     IF (IMP.EQ.1.0) GC TC 10
      Z80=Z80Z1+DEL2
      GC TO 15
10    Z80=Z20
      DEL2=(Z20-Z20Z1)/4.0
15    CONTINUE
      Z20Z1=Z20
      Z8CZ1=Z8C
20    CONTINUE
ENDMAC
*
*     *****************************
*     ******* RATE LIMITER ********
*     *****************************
*
MACRO ROUT=RATLM(RIN,IMP,INC)
  PROCEDURAL
    IF(KEEP.NE.1) GC TO 100
    IF (TIME.NE.0.0) GO TO 80
    ROUTZ1=RIN
80  CONTINUE
    DEL1=RIN-ROUTZ1
    DELLIM=LIMIT(-INC,INC,DEL1)
    ROUTA=ROUTZ1+DELLIM
    IF(IMP.NE.1.0) GC TC 90
    ROUTZ1=ROUTA
90  CONTINUE
    ROUT=ZHOLD(IMP,ROUTA)
100 CONTINUE
ENDMAC
*
INITIAL
*
*     *********************************
*     DIGITAL FILTER INITIAL CONDITICNS
*     *********************************
*
 INCCN PR8Z1=.0,PR8Z2=.0,PR9Z1=.0,PR9Z2=.0,PG3Z1=.0,PG4Z1=.0,...
    RG3Z1=.0,RG3Z2=.0,RG4Z1=.0,RG4Z2=.0,RG4FZ1=.0,RG4FZ2=.0,...
    RG5Z1=.0,RG5Z2=.C,RG5FZ1=.0,RG5FZ2=.0,RG6Z1=.0,RG6Z2=.0,...
    NZAFZ1=.0,NZ4Z1=.0,NZ5Z1=.0,ST7Z1=.0,ST15Z1=.0,ST15Z2=.0,...
    ST16Z1=.0,ST16Z2=.0,TE2Z1=.0,TE2Z1=.0,RSR2Z1=.0,RSR3Z1=.0,...
    NY6Z1=.0,YG6Z1=.0,YG7Z1=.0,YV3BZ1=.0,YV3BZ2=.0,YV3AZ1=.0,...
    YV3AZ2=.0,YV3FZ1=.C,YV3Z1=.0
*
*     *********************************
*     PS,QC,PK,RK,YK AND RATE CCNSTANTS
*     *********************************
*
 CONSTANT PS=1761.0,QC=800.0,PK9=-1.1543,PK10=0.4647,PK12=0.5654,...
    RK1=1.177,RK2=3.22,RK7=50.0,RK10=1.33,RK11=190.0,RK13=1.0,...
    YK7=.51271,YK8=.04877,YK15=1.01266,YK16=.02469,...
    RATE20=0.05,RATE40=0.025,RATE80=0.0125
*
DYNAMIC
*
   SCRT
*
*     ****************
*     *****INPUTS*****
*     ****************
*
      PP1=00.0*STEP(C.0)
```

```
          PR1=0.0*STEP(0.0)-00.0*STEP(2.0)
          PY1=20.0*STEP(0.0)-40.0*STEP(2.0)
          PG1=00.0*STEP(0.0)
          RG1=00.0*RAMP(0.0)-00.0*RAMP(2.0)
          YG1=0.5*RAMP(0.0)-1.5*RAMP(2.0)
          AA1=7.0*STEP(0.0)+0.0*RAMP(0.0)
          NZ1=0.0*STEP(0.0)
          NY1=0.0*RAMP(0.0)-0.0*RAMP(2.0)
*
          RI=QC/PS
          PIQ=LIMIT(0.0,1800.0,PS)
          QKF=LIMIT(200.0,2000.0,QC)
*
*         ***************************
*         *****IMPULSE FUNCTIONS*****
*         ***************************
*
          IMP20=IMPULS(0.0,RATE20)
          IMP40=IMPULS(0.0,RATE40)
          IMP80=IMPULS(0.0,RATE80)
*
*         *************************************
*         *****ANGLE OF ATTACK SENSOR PATH*****
*         *************************************
*
          AA2=CMPXPL(0.0,0.0,0.74,209.0,AA1*43681.0)
          AA3=ZHOLD(IMP40,AA2)
          ALPHAT=0.59*AA3+1.9
          AA6=ALPHAT-22.0
          AA7=LIMIT(0.0,10000.0,AA6)
          PK17=ZHOLD(IMP20,F37(NZA)*F32A(QKF))
          PV1=PK17*AA7
          PV1A=ZHOLD(IMP20,PV1)
*
*         *************************
*         *****RATE GYRO PATHS*****
*         *************************
*
          PG2=CMPXPL(0.0,0.0,0.89,78.5,PG1*6162.25)
          PG3=ZHOLD(IMP80,PG2)
          PK11=1.65*F22(QC)
          P2A=1.0+PK11*(1.0-PK12)
          P2B=(1.0+PK11)*(1.0-PK12)
          P2C=1.0-PK12
          PG4=ZLAG(PG3,P2A,P2B,P2C,IMP80,PG3Z1,PG4Z1)
          PV4=LIMIT(-80.0,120.0,PG4)
          PG5=PV4*((F68(QC)*F32A(QKF))+F40(RI,PS,QC,PIQ))
          PV4A=ZHOLD(IMP20,PV4)
          RG2=CMPXPL(0.0,0.0,0.8,90.0,RG1*8100.0)
          RG3=ZHOLD(IMP80,RG2)
          RG4=ZNOTCH(RG3,.16518,.33036,.16518,-.50084,.16157,IMP80,...
                    RG3Z1,RG3Z2,RG4Z1,RG4Z2)
          RG5=ZNOTCH(RG4,.75892,-1.24831,.70534,-1.24831,.46426,IMP80,...
                    RG4FZ1,RG4FZ2,RG5Z1,RG5Z2)
          RG6=ZNOTCH(RG5,.61223,-.27505,.57751,-.27505,.18974,IMP80,...
                    RG5FZ1,RG5FZ2,RG6Z1,RG6Z2)
          RG6A=DEADSP(-RV16,RV16,RG6)
          RK6=F4(QC,PS)
          RG6B=LIMIT(-10.0,10.0,RG6)*LIMIT(0.0,0.12,.12-RK6)
          RG7=RG6*RK6+RG6A+RG6B
          YG2=CMPXPL(0.0,0.0,0.89,200.0,40000.0*YG1)
          YG3=ZHOLD(IMP40,YG2)
          CSAOAT=1.0-0.5*((ALPHAT*0.0174533)**2.0)
          YG4=YG3*CSAOAT
          SNAOAT=(1.0-0.125*((ALPHAT*0.0174533)**2.0))*ALPHAT*0.0174533
          YG5=YG4-SNAOAT*ZHOLD(IMP40,RG3)
          YG6=RATLM(YG5,IMP40,15.0*RATE40)
          Y3A=(YK15+(1.0-1.9875*YK15)*YK16)
          Y3B=YK15*(1.0-YK16)
          Y3C=1.0-YK16
          YG7=ZLAG(YG6,Y3A,Y3B,Y3C,IMP40,YG6Z1,YG7Z1)
          YG8=RATLM(YG7,IMP40,15.0*RATE40)
*
*         ************************************
*         *****NORMAL ACCELEROMETER PATH*****
*         ************************************
```

```
*
      NZ2=CMPXFL(0.0,0.0,0.89,200.0,NZ1*40000.0)
      NZ3=ZHOLD(IMP40,NZ2)
      NZA=NZ3
      NZAB1=-50.0*LIMIT(0.0,4.0,NZA)+280.0
      RV16=AMIN1(NZAB1,RK11)
      NZ4=NZ3-(RG3**2)*(6.8529E-6)
      NZ4A=ZHOLD(IMP20,NZ4)
      NZAF=ZINT(NZ4A,0.25,0.75,IMP20,NZAFZ1)
      P5A=(1.0+PK9*(1.0-PK10))
      P5B=(1.0+PK9)*(1.0-PK10)
      P5C=1.0-PK10
      NZ5=ZLAG(NZ4,P5A,P5B,P5C,IMP40,NZ4Z1,NZ5Z1)
      PV2=LIMIT(-10.0,10.0,NZ5)
      NZ7=PV2*3.5*F32A(QKP)
      PV2A=ZHOLD(IMP20,PV2)
*
*     ****** ******************************
*     ****LATERAL ACCELEROMETER PATH****
*     ****** ******************************
*
      NY2=CMPXFL(0.0,0.0,0.89,78.54,6168.53*NY1)
      NY3=ZHOLD(IMP40,NY2)
      NY4=NY3+(ZHOLD(IMP40,RG3)*((1.0/347.0))**2.0)*ZHOLD(IMP40,RG3)
      NY5=F9C(RI,PS)*LIMIT(-1.8,1.8,NY4)
      NY6=ZINT(NY5,0.4,0.6,IMP40,NY6Z1)
      NY7=NY5+NY6*F112(PI)*F113(ALPHAT)
*
*     ****** ***************************
*     ********PILOT INPUT PATH********
*     ****** ***************************
*
      PP2=DEADSP(-2.0,2.0,PP1)
      PP3=CMPXFL(0.0,0.0,0.14,27.3,PP2*106.47)
      PP4=REALFL(0.0,7.9365E-3,PP3)
      PP5=ZHOLD(IMP80,PP4)
      PP6=PP5*(7.0+(0.2*ABS(PP5)))
      PP7=LIMIT(-25.0,50.0,PP6)
      PV3=PP7
      PV3A=ZHOLD(IMP20,PV3)
      PR2=DEADSP(-2.0,2.0,PR1)
      PR3=CMPXFL(0.0,0.0,0.7,24.3,PR2*161.04)
      PR4=LIMIT(-3.0,3.0,PR3)
      PR5=REALFL(0.0,0.0079,PR4)
      PR6=ZHOLD(IMP80,PR5)
      PR7=DEADSP(-0.025,0.025,PR6)
      PR8=PR7*(ABS(PR7)*RK1+RK2)
      PR9=ZNOTCH(PR8,.6555,-1.5512,.8234,-1.5512,.6789,IMP80,...
                 PR8Z1,PR8Z2,PR9Z1,PR9Z2)
      PY2=DEADSP(-7.0,7.0,PY1)
      PY3=CMPXFL(0.0,0.0,1.5,66.6,4435.6*PY2)
      PY4=ZHOLD(IMP40,PY3)
      PY5=DEADSP(-1.0,1.0,PY4)
      YK9=0.00072*F10(QC)
      YK10=0.234*F10(QC)
      PY6=PY5*(YK9*ABS(PY5)+YK10)
      YV1=PY6
      PY7=PY6*(0.5-(F17(ALPHAT)*F114(RI)))
      PY8=LIMIT(-30.0,30.0,PY7)
*
*     ************************
*     *****STABILATOR PATH*****
*     ************************
*
      PK19=ZHOLD(IMP20,F12(RI,PS)*F32A(QKP)*3.5)
      PK21=ZHOLD(IMP20,F12(RI,PS)*F32A(QKP)*F68(QC))
      PK22=ZHOLD(IMP20,F12(RI,PS)*F32A(QKP))
      ST1=PV3A*PK22
      ST2=(PV4A*PK21)-ST1
      ST3=(PK19*PV2A)+PV1A+ST2
      ST4=LIMIT(-10000.0,0.0,ST3)
      ST5=LIMIT(0.0,10000.0,ST3)
      ST6=(ST4*F23(ALPHAT))+ST5
      ST8=IMPL(0.0,0.05,ST7)
      ST6A=ST6-0.5*ST8
      ST7=ZINT(ST6A,0.05,1.0,IMP20,ST7Z1)
```

```
      ST9=AV204C(LIMIT(-50.0,25.0,ST8),IMP20)
      ST10=NZ7+ST9
      ST11=ST1C+PV1
      ST12A=ST11+((RG3*YG3)*F107(QC))
      ST12=ZHOLD(IMP40,ST12A)
      ST13=AV4080(ST12,IMP40)
      ST14=ST13+PG5
      ST15=LIMIT(-25.0,25.0,ST14)
      ST16=ZNOTCH(ST15,.69C84,-.99068,.66312,-.99068,.35396,...
           IMP8C,ST15Z1,ST15Z2,ST16Z1,ST16Z2)
      ST17=ST16-(PV3*F32A(QKP))
      ST18=LIMIT(-24.0,10.5,ST17)
      DS1=RV2+RV1A*1.33
      DS2=LIMIT(-50.0,50.0,DS1)
      DS3=DS2*RV7
      ST2R1=17.25-ABS(ST2+6.75)
      ST2R2=LIMIT(-5C.0,F41(ALPHAT,RI)*10.0,ST2R1)
      DS4=LIMIT(-ST2R2,ST2R2,DS3)
      DS5=RATLM(DS4,IMP80,35.0*RATE8C)
      RST1=ST18-DS5
      RST2=LIMIT(-24.0,10.5,RST1)
      RST3=QNTZR(RATE80,RST2)
      RSTDEF=CMPXPL(0.0,0.0,0.7,40.0,RST3*1600.0)
      LST1=ST18+DS5
      LST2=LIMIT(-24.0,10.5,LST1)
      LST3=QNTZR(RATE80,LST2)
      LSTDEF=CMPXPL(0.0,0.0,0.7,40.C,LST3*1600.0)
*
*     ********************
*     ****AILERCN PATH****
*     ********************
*
      RV2A=PR9*((F4(QC,PS)+F13(RI,PS))*F7(QC,PS))-RG7
      RV2=LIMIT(-50.0,5C.0,RV2A)
      RV11=F31(RI,PS)*F34(ALPHAT)
      RK17=F6(RI,PS,RV11)
      RV7=AMIN1(RK17*P35(ALPHAT),RK17-F101(RI,PS,NZAP))
      YV1R=LIMIT(0.0,1.0,F39(ALPHAT))*YV1
      RV1=LIMIT(-200.0,200.0,YV1R)
      RV1A=AV4C80(RV1,IMP40)
      AL1=RV2+RV1A*1.33
      AL2=LIMIT(-50.C,5C.0,AL1)
      AL3=AL2*F35(ALPHAT)*F36(RI,QC,PS)
      AL4=LIMIT(-50.C*F41(ALPHAT,RI),50.0*F41(ALPHAT,RI),AL3)*0.5
      AL5=DEADSP(-42.0,42.0,AL4)+AL4
      RAL1=LIMIT(-42.0,25.0,AL5)
      RAL2=RATLM(RAL1,IMP8C,100.0*RATE80)
      RAL3=QNTZR(RATE80,RAL2)
      RALDEF=CMPXPL(0.0,0.C,0.7,40.C,-1600.0*RAL3)
      LAL1=LIMIT(-25.0,42.C,AL5)
      LAL2=RATLM(LAL1,IMP80,100.0*RATE80)
      LAL3=QNTZR(RATE80,LAL2)
      LALDEF=CMPXPL(0.0,0.C,0.7,40.0,1600.0*LAL3)
*
*     ********************
*     ****RUDDER PATH****
*     ********************
*
      RSR1=LAL2+RAL2+DS5*2.0
      RSR2=ZHOLD(IMP40,RSR1)
      Y5A=(1.0+YK7*(1.0-YK8))
      Y5B=(YK7+1.0)*(1.0-YK8)
      Y5C=1.0-YK8
      RSR3=ZLAG(RSR2,Y5A,Y5B,Y5C,IMP40,RSR2Z1,RSR3Z1)
      RSR4=RSR3*F30(QC,PS)*F38(ALPHAT,RI,PS)
      YV2=F42(RSR4)
      YV3D=ZHOLD(IMP40,RG3)*ZHOLD(IMP40,PG3)*F108(QC)
      YV3C=YG8*F96(QC,PS)+NY7-LIMIT(-30.0,30.0,YV3D)
      YV3B=LIMIT(-30.0,30.0,YV3C)
      YV3A=ZNOTCH(YV3B,.13876,.27752,.13876,-.7914,.34642,IMP40,...
                  YV3BZ1,YV3BZ2,YV3AZ1,YV3AZ2)
      YV3=ZLAG(YV3A,.44837,-.44837,.10326,IMP40,YV3BZ1,YV3Z1)
      RD1=YV2+YV3-PY8
      RD2=RD1*F45(QC,PS)
      RD3=LIMIT(-30.0,30.0,RD2)
      RRD1=LIMIT(-3C.0,30.0,RD3)
```

```
           RRD2=AV4C80(RRD1,IMP40)
           RRD3=QNTZR(RATE80,RRD2)
           RRDDEF=CMPXPL(0.0,0.0,0.7,40.0,1600.0*RRD3)
           LRD1=LIMIT(-30.0,30.0,RD3)
           LRD2=AV4C80(LRD1,IMP40)
           LRD3=QNTZR(RATE80,LRD2)
           LRDDEF=CMPXPL(0.0,0.0,0.7,40.0,1600.0*LRD3)
*
*          ********************************
*          *****LEADING EDGE FLAP PATH*****
*          ********************************
*
           LE2=ZINT(ALPHAT,0.0625,0.9375,IMP40,LE2Z1)
           LE2T=ZHOLD(IMP20,LE2)
           LE2T1=F27(LE2T,RI)
           LE2T2=LIMIT(0.0,F28(QC),LE2T1)
           LE3=LIMIT(0.0,F29U(RI),LE2T2)
           LE4=RATLM(LE3,IMP20,18.0*RATE20)
           LE5=AV2080(LE4,IMP20)
           DLE1=RV2*F93(RI,PS,NZAP)
           DLE2P1=LE5+3.0
           DLE2=DEADSP(-DLE2P1,DLE2P1,DLE1)
           DLE3=DLE1+DLE2
           RLE1=LE5+DLE3
           RLE2=LIMIT(-3.0,33.0,RLE1)
           RLE3=RATLM(RLE2,IMP80,18.0*RATE80)
           RLE4=LIMIT(-3.0,33.0,RLE3)
           RLE5=QNTZR(RATE80,RLE4)
           RLEDEF=CMPXPL(0.0,0.0,0.7,40.0,RLE5*1600.0)
           LLE1=LE5-DLE3
           LLE2=LIMIT(-3.0,33.0,LLE1)
           LLE3=RATLM(LLE2,IMP80,18.0*RATE80)
           LLE4=LIMIT(-3.0,33.0,LLE3)
           LLE5=QNTZR(RATE80,LLE4)
           LLEDEF=CMPXPL(0.0,0.0,0.7,40.0,LLE5*1600.0)
*
*          ********************************
*          *****TRAILING EDGE FLAP PATH*****
*          ********************************
*
           TE2=ZINT(ALPHAT,0.03125,0.96875,IMP40,TE2Z1)
           TE2T=ZHOLD(IMP20,TE2)
           TE2T1=F24(RI,TE2T)
           TE3=LIMIT(0.0,F25(QC),TE2T1)
           TE4=AV2080(TE3,IMP20)
           DTE1=RATLM(RV2*RV11,IMP80,18.0*RATE80)
           RTE1=TE4-DTE1
           RTE2=LIMIT(-8.0,45.0,RTE1)
           RTE3=QNTZR(RATE80,RTE2)
           RTEDEF=CMPXPL(0.0,0.0,0.7,40.0,RTE3*1600.0)
           LTE1=TE4+DTE1
           LTE2=LIMIT(-8.0,45.0,LTE1)
           LTE3=QNTZR(RATE80,LTE2)
           LTEDEF=CMPXPL(0.0,0.0,0.7,40.0,LTE3*1600.0)
TERMINAL
METHOD RKSFX
TIMER FINTIM=4.00,PRDEL=0.25,OUTDEL=0.25,DELT=0.0125
         TITLE RUDDER RESPONSE TO +/-20LB YAW
         PRINT RRDDEF,LRDDEF
         PAGE XYPLOT
*        OUTPUT TIME,RLEDEF(-5.0,5.0),LLEDEF(-5.0,5.0)
*        LABEL DIFF.L.E. FLAP RESPONSE TO +/-10LB ROLL, WITH RG FEEDBACK
*        LABEL M=1.40,H=5000FT
*        OUTPUT TIME,RTEDEF(-5.0,5.0),LTEDEF(-5.0,5.0)
*        LABEL DIFF.T.E. FLAP RESPONSE TO +/-10LB ROLL, WITH RG FEEDBACK
*        LABEL M=1.40,H=5000 FT
         OUTPUT TIME,RRDDEF(-5.0,5.0),LRDDEF(-5.0,5.0)
         LABEL RUDDER RESPONSE TO +/-20LB YAW, WITH YG,NY,RG FEEDBACK
         LABEL M=0.81,H=5000FT
        END
        STOP
           FUNCTION F4(QC,PS)
           REAL LIMIT
           F4T1=0.36-0.0012*QC
           F4T2=LIMIT(200.,1000.,PS)
           F4T3=0.2-0.0003*F4T2
```

```
      F4T4=0.1-0.0001*F4T2
      IF(.NOT.F4T2.LE.500.) GO TO 10
        F4T5=F4T3
        GO TO 20
10    CONTINUE
      F4T5=F4T4
20    CONTINUE
      F4=LIMIT(F4T5,0.3,F4T1)
      RETURN
      END
      FUNCTION F6(RI,PS,RV11)
      REAL LIMIT
      F6T1=0.22*LIMIT(0.0,1.12,RI)+(1.04E-4)*PS
      F6T2=LIMIT(0.0,0.431,F6T1)-0.25*RV11
      F6=.511-LIMIT(0.311,0.431,F6T2)
      RETURN
      END
      FUNCTION F7(QC,PS)
      REAL LIMIT
      F7T1=0.05*LIMIT(25.0,125.0,QC)+3.75
      F7T2=QC-325.0
      F7T3=PS-628.0
      F7T4=(LIMIT(-325.,325.,F7T2)*LIMIT(-386.,0.,F7T3))*(2.71E-5)+10.0
      IF(.NOT.QC.LE.325.0) GO TO 10
        F7=F7T1
        GO TO 15
10    CONTINUE
      F7=F7T4
15    CONTINUE
      RETURN
      END
      FUNCTION F10(QC)
      REAL LIMIT
      F10T1=1.0-0.00042S*QC
      F10=LIMIT(0.35,1.0,F10T1)
      RETURN
      END
      FUNCTION F12(RI,PS)
      REAL LIMIT
      F12T1=RI**2*9.625-.025*RI+1.0
      F12T2=PS*7.969E-4+0.84
      F12MAX=LIMIT(1.0,8.0,F12T2)
      F12T3=LIMIT(1.0,F12MAX,F12T1)
      F12T5=LIMIT(.5,1.35,RI)
      F12T6=F12T5*(0.00952*PS+4.04) +(-PS*0.00396-1.18)
      F12T4=LIMIT(1.0,8.0,F12T6)
      IF(.NOT.RI.GT.0.5) GO TO 5
      F12 = F12T4
      GO TO 6
5     CONTINUE
      F12 = F12T3
6     CONTINUE
      RETURN
      END
      FUNCTION F13(RI,PS)
      REAL LIMIT
      PS13L=LIMIT(200.0,2116.0,PS)
      F13T1= (445.0+LIMIT(242.0,628.0,PS13L))*(2.39E-4)
      IF(.NOT.PS13L.LE.800.0) GO TO 10
          F13T2A=0.054+PS13L*(3.59E-4)
          F13T2B=-0.1745-PS13L*(8.4E-5)
        GO TO 20
10    CONTINUE
      F13T2A=0.00122*PS13L-0.637
      F13T2B=0.192-PS13L*(5.42E-4)
20    CONTINUE
      F13T2C=0.152+PS13L*(5.34E-5)
      RI13L=LIMIT(0.0,2.8,RI)
      F13T3= F13T2C+RI13L*(F13T2B+RI13L*F13T2A)
      F13=LIMIT(0.13,F13T1,F13T3)
      RETURN
      END
      FUNCTION F17(ALPHAT)
      REAL LIMIT
      F17T1=ALPHAT-13.0
      F17=LIMIT(0.0,4.0,F17T1)
```

```
      RETURN
      END
      FUNCTION F22(QC)
      REAL LIMIT
      F22=0.0167*(-800.0+LIMIT(800.0,900.0,QC))
      RETURN
      END
      FUNCTION F23(ALPHAT)
      REAL LIMIT
      F23=3.1435-0.1429*LIMIT(15.0,21.998,ALPHAT)
      RETURN
      END
      FUNCTION F24(RI,ALPHAT)
      REAL LIMIT
      F24L1=22.538-20.51*LIMIT(0.27,0.66,RI)
      F24L2=32.76-36.0*LIMIT(0.66,0.91,RI)
      F24T1=LIMIT(0.0,10000.0,ALPHAT)
      F24T2=ALPHAT-(14.8769-7.6923*LIMIT(0.27,0.91,RI))
      F24T3=-2.0*LIMIT(0.0,10000.0,F24T2)
      F24T4=1.4*(F24T1+F24T3)
      IF (.NOT.RI.GT.0.66) GO TO 10
         F24L=F24L2
      GO TO 15
10    CONTINUE
         F24L=F24L1
15    CONTINUE
      F24=LIMIT(0.0,F24L,F24T4)
      RETURN
      END
      FUNCTION F25(QC)
      REAL LIMIT
      F25=47.636-0.05106*LIMIT(600.0,835.0,QC)
      RETURN
      END
      FUNCTION F27(ALPHAT,RI)
      REAL LIMIT
      F27=1.328*(ALPHAT+7.8584-17.86*LIMIT(0.44,0.63,RI))
      RETURN
      END
      FUNCTION F28(QC)
      REAL LIMIT
      F28=44.551-0.04058*LIMIT(260.0,950.0,QC)
      RETURN
      END
      FUNCTION F29U(RI)
      REAL LIMIT
      F29U=87.3825-76.25*LIMIT(0.7,1.146,RI)
      RETURN
      END
      FUNCTION F30(QC,PS)
      REAL LIMIT
      F30T1=1.0803-(LIMIT(140.0,800.0,QC))*0.0012878
      F30T2=1.7035-(LIMIT(242.0,498.0,PS))*0.0033203
      F30=LIMIT(F30T2,0.9,F30T1)
      RETURN
      END
      FUNCTION F31(RI,PS)
      REAL LIMIT
      F31T1=(LIMIT(0.0,2116.0,PS))*(6.536E-5)
      F31T2=1.449-(LIMIT(1455.0,2116.0,PS))*(5.1E-4)
      IF(.NOT.RI.LE.0.717) GO TO 10
         F31T3=F31T2
         GO TO 20
10    CONTINUE
      F31T3=0.112
20    CONTINUE
      F31T4=RI-0.717
      F31T5=(LIMIT(-0.717,0.662,F31T4))*F31T3
      F31T6=0.2247-F31T1-F31T5
      F31=LIMIT(0.0,0.16,F31T6)
      RETURN
      END
      FUNCTION F32A(QKF)
      F32A=100.0/QKF
      RETURN
      END
```

```
      FUNCTION F34(ALPHAT)
      REAL LIMIT
      F34T1=1.5-0.2*ABS(ALPHAT-2.5)
      F34=LIMIT(0.0,1.0,F34T1)
      RETURN
      END
      FUNCTION F35(ALPHAT)
      REAL LIMIT
      F35T1=0.955-0.0325*ALPHAT
      F35T2=0.5+0.04221*ALPHAT
      IF(.NOT.ALPHAT.LE.6.0) GO TO 10
         F35T3=F35T2
         GO TO 20
10    CONTINUE
      F35T3=F35T1
20    CONTINUE
      F35=2.0*LIMIT(0.175,0.5,F35T3)
      RETURN
      END
      FUNCTION F36(RI,QC,PS)
      REAL LIMIT
      F36T1=LIMIT(0.0,0.8,RI)+PS*(1.69E-4)-.788
      F36T2=(440.0-LIMIT(440.0,640.0,PS))*7.46E-3
      F36T3=(LIMIT(0.7,1.9,RI))*540.0+955.0-QC
      F36T4=(LIMIT(0.0,200.0,F36T3))*0.005
      F36T5=F36T1*F36T2+1.0
      F36T6=LIMIT(0.0,1.0,F36T5)
      F36=AMIN1(F36T4,F36T6)
      RETURN
      END
      FUNCTION F37(NZA)
      REAL LIMIT
      F37=2.5-0.5*LIMIT(3.0,5.0,NZA)
      RETURN
      END
      FUNCTION F38(ALPHAT,RI,PS)
      REAL LIMIT
      FSRI=LIMIT(0.225,0.549,RI)
      IF(.NOT.ALPHAT.LT.16.0) GO TO 10
         FS=-0.24308*FSRI+0.03967
         AOAM=-10.0
         AOAB=-0.45
         GO TO 20
10    CONTINUE
      FS=-0.13846*FSRI-0.01774
      AOAM=-5.467
      AOAB=-0.025
20    CONTINUE
      F38T1=LIMIT(-5.0,20.4,ALPHAT)
      F38T2=LIMIT(-5.0,25.0,ALPHAT)
      IF(.NOT.RI.LE.0.334) GO TO 30
         F38T3=F38T2
         LL=-0.98
         GO TO 40
30    CONTINUE
      F38T3=F38T1
      LL=-1.2
40    CONTINUE
      F38T4=(F38T3+AOAM)*FS+AOAB
      F38T5=LIMIT(LL,0.3,F38T4)
      F38T6=0.048-0.0498*LIMIT(-5.0,25.0,ALPHAT)
      F38T7=LIMIT(-5.0,25.0,ALPHAT)+0.01*LIMIT(240.0,390.0,PS)-3.9
      F38T8=LIMIT(0.0,2.5,F38T7)-2.5
      F38T9=RI-0.72
      F38T10=0.0168*LIMIT(0.0,2.08,F38T9)
      F38T11=F38T10*F38T8+F38T6
      IF(.NOT.ALPHAT.LE.10.0) GO TO 50
         F38=F38T11
         GO TO 60
50    CONTINUE
      F38=F38T5
60    CONTINUE
      RETURN
      END
      FUNCTION F39(ALPHAT)
      REAL LIMIT
```

```
      F39=(LIMIT(13.0,25.0,ALPHAT))*0.08333-1.08329
      RETURN
      END
      FUNCTION F40(RI,PS,QC,PIQ)
      REAL LIMIT
      F40T1=3.25-3.0*LIMIT(0.75,0.85,RI)
      F40T2=0.65625-.0013125*LIMIT(500.0,1800.0,PS)
      F40T3=-0.26177+(9.635E-4)*LIMIT(500.0,1800.0,PS)
      F40T4=LIMIT(0.0,105.0,QC-335.0)
      F40T5=(F40T4*(-PIQ*(1.17428E-6)+(1.5238E-3)))+0.475-(6.5E-4)*PIQ
      F40T6=F40T3+(F40T2*LIMIT(0.75,0.85,RI))
      F40T7=F40T4*(5.6746E-4+1.9841E-7*PIQ)+(0.16923-3.84615E-5*PIQ)
      F40T8=(0.16923-3.86415E-5*PIQ)+(F40T4*(1.67247E-3-9.29152E-7*PIQ))
      IF (.NOT.RI.GE.0.75) GO TO 20
         F40T9=F40T8*F40T1
         GO TO 30
20    CONTINUE
         F40T9=F40T8
30    CONTINUE
      IF (.NOT.(QC.GE.440.0.AND.RI.GE.0.75)) GO TO 40
         F40T10=F40T6
         GO TO 50
40    CONTINUE
         F40T10=F40T7
50    CONTINUE
      IF (.NOT.PIQ.GT.980.0) GO TO 60
         F40=F40T9
         GO TO 80
60    CONTINUE
      IF (.NOT.PIQ.LE.500.0) GO TO 70
         F40=F40T5
         GO TO 80
70    CONTINUE
         F40=F40T10
80    CONTINUE
      RETURN
      END
      FUNCTION F41(ALPHAT,RI)
      REAL LIMIT
      F41T1=(LIMIT(8.0,22.0,ALPHAT)-8.0)*0.053571
      F41T2=(.23533*LIMIT(.27,.48,RI)-.00997)*(LIMIT(8.,22.,ALPHAT)-8.)
      F41T4=0.5875-(LIMIT(0.27,0.39,RI))*1.25
      IF (.NOT.ALPHAT.LE.8.0) GO TO 10
         F41T3=1.0-F41T1
         GO TO 20
10    CONTINUE
      F41T3=1.0-F41T2
20    CONTINUE
      F41=LIMIT(F41T4,1.0,F41T3)
      RETURN
      END
      FUNCTION F42(IN)
      REAL LIMIT, IN
      F42=(ABS(LIMIT(-18.33,18.33,IN))*0.0479+0.761)
     *    *LIMIT(-18.33,18.33,IN)
      RETURN
      END
      FUNCTION F45(QC,PS)
      REAL LIMIT
      F45T1=(LIMIT(500.0,1000.0,QC)-500.0)*0.001
      F45T2=(LIMIT(625.0,1450.0,PS))*0.0012121-0.75758
      F45=1.0-F45T1*F45T2
      RETURN
      END
      FUNCTION F68(QC)
      REAL LIMIT
      F68=-0.002977*(-480.0+LIMIT(260.0,480.0,QC))
      RETURN
      END
      FUNCTION F90(RI,PS)
      REAL LIMIT
      F90T1=628.0-PS
      F90T2=RI-0.72
      F90=(LIMIT(0.0,386.0,F90T1)*LIMIT(0.0,2.08,F90T2))*0.020551+16.5
      RETURN
      END
```

```
      FUNCTION F93(RI,PS,NZAF)
      REAL LIMIT,NZAF
      F93T1=(RI+0.00018*PS-0.69)*0.42857
      F93T2=PS-628.0
      F93T3=1.0-0.1*LIMIT(0.0,5.0,NZAF)
      F93T4=LIMIT(0.0,0.06,F93T1)
      F93T5=(6.3694E-3)*LIMIT(0.0,157.0,F93T2)
      F93=F93T4*F93T5*F93T3
      RETURN
      END
      FUNCTION F96(QC,PS)
      REAL LIMIT
      F96T1=1.54-0.0044*LIMIT(0.0,168.0,QC)
      F96T2=QC*0.00088+0.36
      IF(.NOT.QC.LE.500.0) GO TO 10
         F96T3=F96T1
         GO TO 20
10    CONTINUE
      F96T3=F96T2
20    CONTINUE
      F96T4=0.0007772*(628.0-LIMIT(242.0,628.0,PS))
      F96T5=F96T3+F96T4
      F96=LIMIT(0.0,1.1,F96T5)
      RETURN
      END
      FUNCTION F101(RI,PS,NZAF)
      REAL LIMIT,NZAF
      F101T1=RI-0.58
      F101T2=LIMIT(0.0,5.0,F101T1)
      F101T3=PS-498.0
      F101T4=F101T2*LIMIT(0.0,835.0,F101T3)
      F101T5=798.54-0.35948*PS
      F101T6=LIMIT(165.3,275.5,F101T5)
      F101T7=(5.4446E-5)*LIMIT(0.0,F101T6,F101T4)
      F101T8=ABS(NZAF)
      F101T9=LIMIT(0.0,5.0,F101T8)
      F101=F101T7*F101T9
      RETURN
      END
      FUNCTION F107(QC)
      REAL LIMIT
      F107T1=((-5.714E-7)*ABS(QC-750.0))+(8.4E-4)
      F107=LIMIT(0.0,1000.0,F107T1)
      RETURN
      END
      FUNCTION F108(QC)
      REAL LIMIT
      F108T1=LIMIT(50.0,390.0,QC)
      F108=(F108T1**2.0)*7.48E-8+0.01557-F108T1*6.435E-5
      RETURN
      END
      FUNCTION F112(RI)
      REAL LIMIT
      IF(RI.LT.0.28.OR.RI.GT.0.76) GO TO 10
         F112=4.0
         GO TO 20
10    CONTINUE
      F112=0.0
20    CONTINUE
      RETURN
      END
      FUNCTION F113(ALPHAT)
      REAL LIMIT
      F113=0.16667*LIMIT(12.0,18.0,ALPHAT)-2.0
      RETURN
      END
      FUNCTION F114(RI)
      REAL LIMIT
      F114=0.89286*LIMIT(0.16,0.30,RI)-0.14286
      RETURN
      END
ENDJOB
```

LIST OF REFERENCES

1. Carter, J., Computer Program to Simulate Digital Computer Based Longitudinal Flight Control Laws in a High Performance Aircraft, Master's Thesis, Naval Postgraduate School, December 1983.

2. Raithel, A., Development of a Flight Simulation Concept and Aerodynamic Buildup for an Investigation of Departure Prevention Systems in Tactical Aircraft, Master's Thesis, Naval Postgraduate School, December 1983.

3. Continuous Systems Modeling Program III (CSMP III), Program Reference Manual, IBM Manual No. SH19-7001, Program No. 5734-XS9, The IBM Company, 1972.

4. Speckart, F., and Green, W., A Guide to Using CSMP, 1st Ed., Prentice-Hall, Inc., 1976.

5. Groll, D., F/A-18 Flight Control System Design Report, Vol. 1, MDC A7813, McDonnell Aircraft Company, December 1982.

6. Hess, R., F/A-18 Flight Control Electronic Set Control Laws, Vols. I and II, MDC A4107, McDonnell Aircraft Company, March 1982.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93943	1
4.	Dr. Marle D. Hewett, Code 67Hj Department of Aeronautics Naval Postgraduate School Monterey, California 93943	5
5.	LCDR James R. Carter, USN 1389 Valencia Loop Chula Vista, California 92010	1
6.	LT Scott F. Graves, USN 4854 W. Glenhaven Drive Everett, Washington 98201	2